实用第一 智慧密集 Practicability First, Intelligence Intensive

电脑编程技巧与维护

COMPUTER PROGRAMMING SKILLS & MAINTENANCE

http://www.comprg.com.cn

上 7月 2013年7月03日

每期定价:11.00元 全年定价:264.00元
《电脑编程技巧与维护》杂志社出版
刊号: ISSN 1006-4052 / CN11-3411/TP
广告许可证 京海工商广字0151

国家级科技期刊 中国学术期刊综合评价数据库统计源期刊 中国核心期刊（遴选）数据库收录期刊

www.directui.com
DirectUI界面库

免费咨询热线：400-660-9989

——让界面与业务逻辑彻底分离

易学易用、缩短80%的界面开发周期、提升界面运行效果与质量

成功应用在华为、中兴、盛大网络、中国移动、铁道研究院、瑞星、步步高等知名企业

第95页.《基于DirectUI架构的中海油软件平台的用户体验设计与实现》



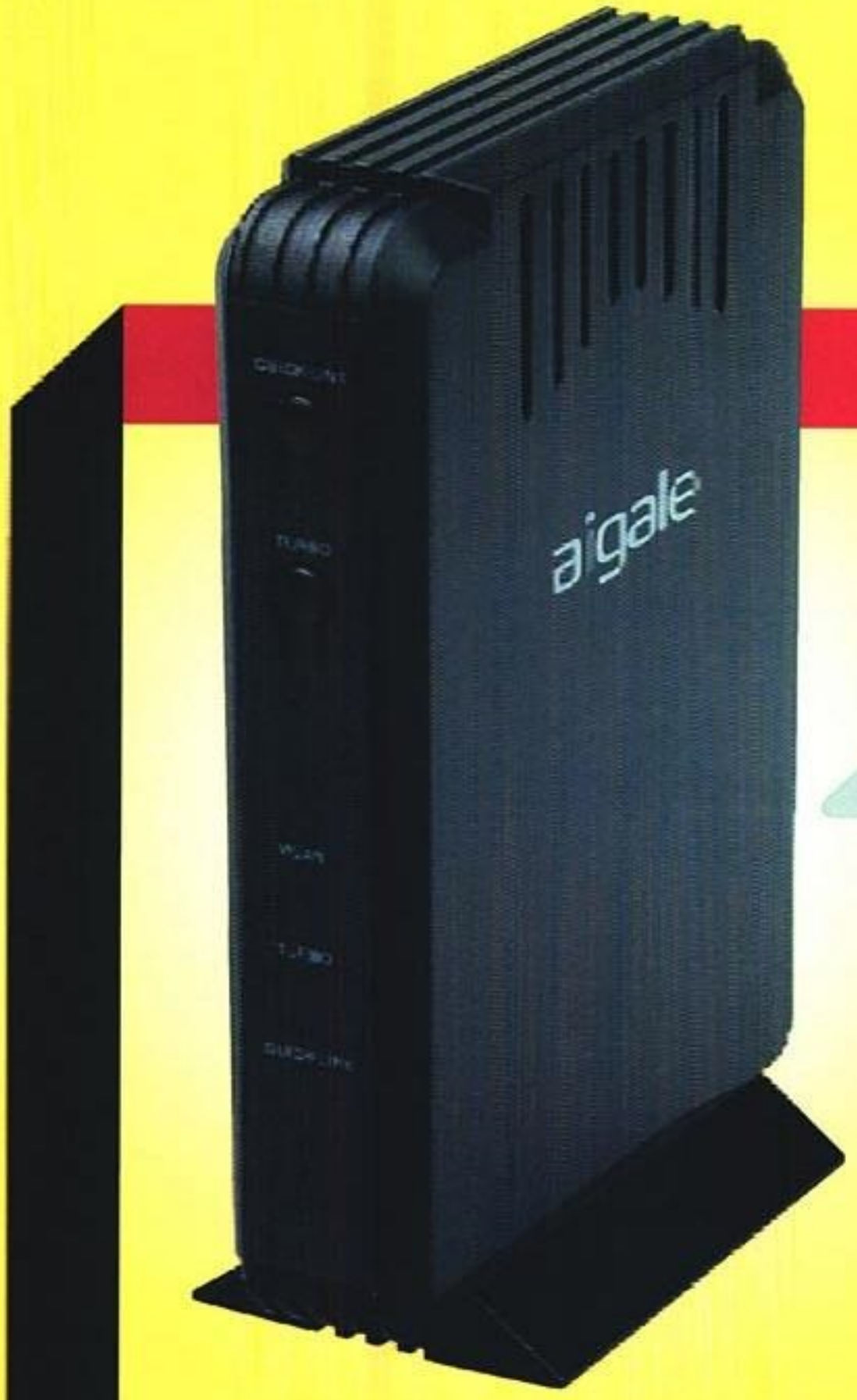
UIPower
www.uipower.com

ISSN 1006-4052

LAICAR.COM
shop35833438.taobao.com

抢先Hold住PCWorld

即可精巧“联”通科技未来！



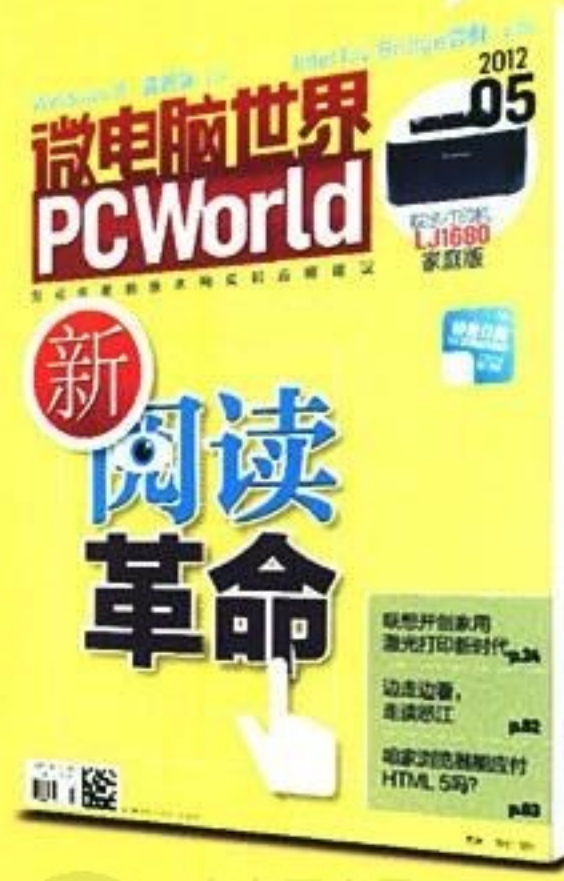
现在邮购2013年
《微电脑世界》**全年杂志**

即得一个价值149元

海联达Ai-R100 极风
无线路由器

轻松联通您的智能终端，
让您尊享全球IT资深顾问
随时随地的贴身资讯服务。

飞信 Fetion
用户可登陆：
http://sms.feixin.10086.cn/Subscribe/getInfo/id/215 订阅
中国移动用户发送ZZ到125200915，即可订阅



汇款地址： 北京市123信箱，收款人：微电脑世界杂志，邮编：100036
杂志定价： 144元/年（12元/月）
活动咨询： 周一到周五，9：00~11：30，13：00~17：30
电话： 010-88230131
杂志社现场订阅地址： 北京市海淀区万寿路翠微中里14号楼
在线订阅： http://www.pcworld.com.cn/about/ebuy/pay.html

活动说明：
活动时间： 2012年8月10日~2013年6月30日（邮局汇款以邮戳为准）
在汇款单附言栏注明“2013年微电脑世界”，同时留下联系电话。
如需发票，请在汇款单附言注明“发票”以及发票抬头，过后将不能补开。
本活动仅限于在杂志社订阅的读者，邮局订阅等其他渠道不参加此活动。
由于本次活动涉及奖品发放，参与活动的读者将不能中途退订。
邮费： 平寄邮费由杂志社承担，如需挂号，每本另加3元挂号费，汇款时一并汇上，并注明挂号字样。

来卡网出品
shop35833438.taobao.com

2013 年第 13 期
7 月(上)

电脑编程技巧与维护

总第 283 期　　1994 年 7 月创刊　　(半月刊)

社　　长：孙茹萍
副 社 长：田　真
总　　编：王路敬
编辑委员会
主　　任：梁祥丰
委　　员：胡顺增　刘　江　莫亚柏
(拼音为序)　孙春亮　温莉芳　吴淑珍
严晓舟　张立荣
编　　辑：侯穆蕾　姬振伟
刘艳彬　杨月慧
美　　编：范志飞
公关部主任：苏加友
出版发行部：刘文海
编 辑 出 版：《电脑编程技巧与维护》杂志社
主 管 部 门：中华人民共和国工业和信息化部
主 办 单 位：中国信息产业商会
地　　址：北京市海淀区长春桥路 5 号
6 号楼 1209 室
投 稿 信 箱：gaojian@comprg.com.cn
gaojian@comprg.sina.net
编辑部信箱：gaojian@comprg.com.cn
发行部信箱：zzsfx@vip.sina.com
网　　址：http://www.comprg.com.cn
邮　　编：100089
电　　话：(010) 82561037
传　　真：(010) 82561614
照　　排：《电脑编程技巧与维护》
杂志社电脑排版部
印 刷 厂：北京慧美印刷有限公司
订 阅 处：全国各地发行局
国内总发行：北京报刊发行局
邮 发 代 号：82-715
国外发行代号：M6232
刊　　号：ISSN 1006-4052 / CN11-3411/TP
广告订可证：京海工商广字 0151 号
全 年 定 价：264 元
每 期 定 价：11 元

飞天诚信
飞天ROCKEY加密锁
引领“智能●低价”风暴
● 震撼价格，超高性价比
● 智能卡芯片
● 无驱，使用更方便
● 涵盖高、中、低端产品


系统支持：
Windows 98SE/Me/2000/XP/Server 2003/2008/Vista/7/8、Linux、*MacOS等多平台
飞天诚信科技股份有限公司　www.FTsafe.com
地址：北京市海淀区学清路9号汇智大厦B楼17层　邮编：100085　电话：010-62304466　传真：010-62304477
华南营销中心：020-38870851　华东营销中心：021-58202268　西南营销中心：028-85481711　华中营销中心：027-87160151

域天32位智能卡


专为共享软件作者设计，使得共享软件作者实现零成本加密！
● 硬件32位智能卡(内置32位CPU)及专有防克隆技术;保证无法复制
● 软件代码在智能卡中运行，内置硬件3DES及RSA算法，无法破解
● 全速USB协议，传输速度高达12Mbps
● 先进的动态加密技术，加密代码不受长度限制
● 支持多种开发语言，在加密锁中可以运行跳转，比较，循环，查表，函数调用等指令及字符串操作
● 超大容量内部储存器：30K字节独立储存空间
● 易于使用的编译及调试器，专有的代码生成器及模糊解释语言，方便开发商进行开发
● 内置时间模块，支持时间限制功能
● 授权锁模式，使得软件的代理销售更容易控制
东莞市域之天软件开发有限公司
电话：0769-22686137 传真：0769-22688320
Http://www.dgyzt.com
E-mail: ytkj_911@163.com

来卡网出品
LAICAR.COM
shop35833438.taobao.com

电脑编程技巧与维护(半月刊)

每月3日出版

实用第一 智慧密集

新技术追踪

4 通过 WiFi 手势识别 有望实现家电操控等三篇

跟高手学编程

5 写给今天的初学者，明天的领航人 …… 唐 洁

10 利用语义网技术实现铁路交通的地理语义查询（二）…… 董 志

根据 SQLite 关系数据库中的数据创建火车站点本体，并生成 OWL 文件的方法和编程实现。

编程语言

16 不会编译原理也能做 Android 计算器 …… 李 宁

结合 JavaCC 自动生成可以解析表达式的 Java 代码，实现 Android 版的计算器。

22 模拟退火算法求解二次规划问题与实现 …… 王文举

介绍了模拟退火算法的基本原理和设计思路，以及在求解二次规划问题中的具体应用。

24 利用 VB 开发 Flash Cookies 清除器 …… 华 云

利用 VB 和 Windows API 函数，及 FSO 文件系统对象，解析开发 Flash Cookies 清除器的编程实现。

27 VBA 编程实现燃气-蒸汽联合循环性能计算 … 卞晶马贤马勇

通过对 Excel VBA 的用法的详细分析，并利用 VBA 编写了计算燃气-蒸汽联合循环性能计算的代码及窗体文件。

29 Android 手机应用开发之信息平台移动客户端 …… 汪永松

从开发者的角度对 Android 手机平台较为流行的信息平台移动客户端进行了全面的功能分析，并通过实例，详细地介绍了此类应用的开发过程。

专家论坛

38 最优数字分配策略的分析与实现之一 …… 尉鹏博 韩银锋

分析了在软件开发过程中最优数字分配策略与实现方法。

数据库

41 基于 Excel VBA 的会计单据演示训练系统开发（三）…… 何 燕

运用 Excel 2007，在 Excel VBA 平台下，开发了会计原始凭证训练演示系统。

46 值班室信息查询系统设计与实现 …… 刘 烽

针对目前企事业单位值班室信息查询工作日趋繁重的现实矛盾，设计并实现了一款功能强大，使用简单的值班室信息查询系统。

网络与通信

58 串口通信新模型的研究与 C# 实现 …… 郑 武 肖宝森

基于完成端口模型，利用 C# 设计出了一个协议性串口通信模块，以封装通信底层的细节，从而提供了方便的通信接口。

60 移动平台下天气信息查询软件的开发 …… 王浩亮

在对当前智能手机系统分析、研究的基础上，开发出了一款基于 Android 平台的天气信息查询软件。

63 Android 利用 SOAP 协议调用天气预报服务 …… 庞国明

基于 Android 平台和 Soap 协议，设计实现了一种实用的手机天气预报软件系统。

67 基于 WAMP 的在线评测系统设计与实现 …… 李臣龙 鲍广喜

详细阐述了在线评判系统的结构原理，并运用 PHP、MySQL 和 Apache 组合，设计实现了基于 B/S 结构的在线评测系统。

图形图像处理

69 GIF 与 BMP 互转程序开发 …… 江 洪

结合对 GIF 与 BMP 文件的结构的分析，使用 VC6.0 开发了一个格式互转的应用程序。

稿件一经采用，即寄样刊，本刊图、文版权归杂志社所有，未经允许不得任意转载和摘编。本刊已许可中国学术期刊（光盘版）电子杂志社在中国知网及其系列数据库产品中以数字化方式复制、汇编、发行、信息网络传播本刊全文，作者如不同意将文章入编，投稿时敬请说明。

来卡网出品
LAICAR.COM
shop35833438.taobao.com

目次



74 Vega Prime 之 SpeedTree 模块程序设计
..... 赵常寿 张鹏 吴红权
结合实例，介绍了 Vega Prime 中SpeedTree 模块程序设计方法，在此基础上可以开发 SpeedTree 模块的具体应用。

游戏编程

77 基于 DI-Guy 的单兵游戏动作设计与实现
..... 陶阳
介绍了 Vega 和 DI-GUY 的基本知识，在对单兵游戏动作进行分类研究的基础上，设计并实现了几个简单的单兵游戏动作。

计算机安全与维护

84 C# 语言实现从 Word 文档中提取文本 吴春龙
C# 编程实现了从 Word 文档二进制文件数据中，过滤掉 Word 文档中的各种格式控制等数据，提取 Word 文档中用户关心需要的文本。

88 XPE 嵌入式设备的远程在线调试 仲光亮
针对使用微软多数的应用程序和开发环境，分析并解决了实现 XPE 嵌入式设备远程在线调试的相关问题。

编程疑难问题解答

90 如何在 Windows 中提升应用程序权限与 UAC 交互 李斌
91 如何快速清除 Word 文档中的重复文章 王昌红

博士信箱

94 电脑系统维护经验与技巧

为您服务

96 新书点评

实用第一 智慧密集

敬告读者：邮政部门独家代理发行本刊，未委托小蓝帽发行公司及其他社会公司办理本刊订阅业务。特此声明！

来卡网出品
LAICAR.COM
shop35833438.taobao.com

通过 WiFi 手势识别 有望实现家电操控

现有的手势识别设备比如 Kinect，通常采用摄像头或者其他的深度传感器捕捉用户动作。而这些设备一般只能在传感器所在的范围感知我们的手势动作。另外，也有通过用户随身携带传感器来实现动作的识别。

而华盛顿大学的计算机专家们刚刚研发的一款设备打破这种限制。他们的技术是采用改进的 WiFi 路由器来识别手势——随便你是在家里的任何位置，只要有 WiFi 信号，设备就能检测到你的手势。

这项技术名为 WiSee，用户可以通过它在家中的任何位置控制所有接入的家电设备，而这一切你要做的只是一个简单的手势。

Wisee 的这套“全方位识别系统”通过分析环境中的 WiFi 信号变化来捕捉用户，而不需要你佩戴任何的外设或者是在家部署一堆摄像头。WiFi 信号还可以穿墙，也就是说你需要配置的只是连接路由器的 WiFi 接入点和你的移动设备。

WiSee 的手势识别是通过捕捉微弱的多普勒效应变化以及人体在环境中移动导致的 WiFi 信号变形实现的。这里多普勒效应的变化是指波源跟接收器的相对位移变化会导致波的频率变化——跟救护车在你身边呼啸而过时的场景非常相似。而对于 WiSee 来说，人体反射回的 WiFi 信号会被接收器当成波源。也就是说当人体在 WiFi 的覆盖中做出某个动作时，那接收器就能获取到对应的多普勒效应的变化。他们只要将不同手势跟不同的多普勒效应变化对应起来就可以了——比如手臂画个圈或者各隔空推两下。

WiSee 实现中的一个问题是：通常的手势动作只会引起非常小幅的多普勒效应变化——大概只有几赫兹的变化——对于将的 WiFi 设备来说，这是非常难捕捉到的。研究人员首先将 WiFi 路由器改造成能够监测家中所有无线信号的智能接收器，之后再在接收器中加入将宽带信号转换成窄带信号的算法，这样设备就能检测到人体手势动作导致的微弱的信号频率变化了。

WiSee 的另一项难度是当检测范围内有多个用户时如何处理。它的解决方法是设定一个特定的触发手势，也就是用户可以通过做出某个特定的动作组合来接管系统的控制权限。研究人员分别在办公和居家的环境中对 5 个用户做过测试，当环境中有多个用户时，手势的识别率会降低。

在类似的研究领域也有微软参与的 SoundWave 和 Humantenna 两个项目：前者采用声音，后者采用电线的辐射来捕捉身体的手势。但 WiSee 是唯一一个可以穿墙工作的系统。

人类记忆将实现上传和下载

人类的记忆向 PC 中的数据一样上传并存储，在之前的科幻片中或许见过，现在科学家告诉你：在未来的数十年里，神经科学、计算机和人工智能的共同发展将不断减小人类大脑和计算机技术之间的隔阂。未来，我们的记忆将通过特定的方式得以存储和提取。

当前，美国出 10 亿美金资助“大脑地图”绘制以及欧盟 13 亿投入在硅基质中打造人类大脑的计划也无疑会帮助加速这一过程。但是，在我们真正实现大脑数据的上传之前，神经学家们表示他们需要首先定位记忆存储的位置和方式。

Ted Berger，一位 USC（南加大）的神经学家表示：“我们已经开始了在复制大脑特性上的探索，同时我们也开始着手如何复制大脑特定区域的问题。”

MIT 的另一拨人马，Ed Boyden 带领的合成神经学团队则在研究能够探索大脑的新工具。

在近年中，他们发现了一种藻类蛋白能将光转化为电能。如果将这种蛋白植入到神经元中，那么后者就可以通过光来触发它的活动。由此神经元的活动就能以电脉冲的形式表现出来尔后编码成计算机能识别的记忆代码。

同时，通过光电蛋白来控制神经元的活动，Boyden 希望能很快研发能够控制脑细胞活动的开关来帮助脑紊乱患者。

科学家研发能用意念绘图的计算机系统

海德·普兹纳 (Heide Pfützner) 居住在德国莱比锡，先前是一名教师。她被诊断患有肌萎缩性脊髓侧索硬化症 (ALS)，这种病有时也被称作“运动神经元病”。这种病会严重影响人体的正常运动机能，然而借助一种新型的人脑操控计算机系统，她仍然可以绘制出精美的画作。

目前这一系统的组成包括一个内置电极的帽子，这个帽子是用来检测佩戴者在思考时的脑电波变化的。

海德·普兹纳表示：她从来没有对技术产品有过什么兴趣，甚至鄙视借助计算机进行工作的做法。但是计算机图形交互界面技术是一项突破，让她得以用意念来进行艺术创作。集中精力和思维可以创作出有意义的图像。

贾维·罗德里格斯 (Javi Rodriguez) 是 gTech 公司的一名工程师，一直从事这套绘画系统的开发工作，他表示自己相信这款软件还可以被用于判断病人是陷于昏迷状态还是处于某种身体的“锁定状态”，即他们感觉自己的身体无法动弹但是思维还是清醒的。

他说：“一般来说非常难以判断病人究竟是已经昏迷还是处于身体上的锁定状态。因此常规的做法是进行一些测试来观察病人的状态。医生会通过耳机或触碰方式向病人发出一些特定的指令并观察他们脑电波的反应。在这一过程中，你所采用的方式不同将对测试结果造成很大的差异。比如说如果病人见到自己的家人，情况就会很不一样，因为如果他们能够听到亲人的声音，但是自己却不能说话，他们的大脑一定会有强烈的反应。通过这种方式还可以帮助病人与他们的家人交流。”

科学家们现正在研发大脑植入技术，以便能够开发出敏感度更高，更加容易使用的升级系统。

来卡网出品 LAICAR.COM shop35833438.taobao.com

编者按：

为了让更多的电脑编程爱好者、应用软件开发人员、系统使用和维护人员分享编程高手的宝贵编程经验、心得体会，展现其编程人生的苦乐年华，述说编程中的感人故事，促进正在学习编程的电脑爱好者与具有丰富编程实践经验的项目研发人员的交流、互动，学习编程高手们新颖的编程理念、编程方法、技巧，以及计算机应用研发经验，本刊于2012年10月1日开始至2013年2月1日期间举办了《编程人生》征文比赛。

本次征文比赛，历时4个月，在广大读者、作者参与和大力支持下，共收到参赛作品200余篇，经多位专家及评审老师的严格评审，评选出了8篇优秀作品。从本期开始，将在本刊上半月刊陆续刊登大赛中评选出的优秀作品，与读者共享编程高手的编程人生。

写给今天的初学者，明天的领航人

唐 洁

作者简介

唐洁（1979－），2001年6月毕业于中国人民解放军炮兵学院计算机应用（软件方向）专业，本科学历。九三学社社员、铜陵市第八届政协委员，现任安徽省铜陵市财政局信息中心主任，工程师职称。有多篇技术论文发表，主持研发了多款软件、游戏引擎及相关应用项目并获奖，研究成果《组合数学Poker分支算法逻辑之公式化模拟》推荐Wizardofodds组合数学协会。目前主要关注数学分析及算法模型设计，D3D游戏引擎设计，协同FD动漫及CA数学协会共同研究开发国产游戏引擎，并一直关注致力于软件自主创新研发领域。

2012是个极特殊的年份，玛雅历新纪元的开始，祖国的改革开放进入全新历史时期，科学发展观正式写入党章。党的十八大报告就明确指出，坚持走中国特色新型工业化、信息化、城镇化、农业现代化道路，推动信息化和工业化深度融合。加快推进国防和军队现代化，加紧完成机械化和信息化建设双重历史任务，力争到2020年基本实现机械化，信息化建设取得重大进展。坚定不移把信息化作为军队现代化建设发展方向，推动信息化建设加速发展。这一切的一切都是那么令人振奋，我虽然从事软件研发12年了，从一线编程一直走到架构师及今天的管理人员，虽说积累了一些经验，也取得了研发专家头衔，但自我感觉还是很惭愧，远没有达到心中的预期水平。我不能算是什么编程专家，只能说积累了一些从业经验而已，从几个方面谈谈我这十几年编程生涯的一些感悟。

1 树立正确的观念，合理选择专业

从高考结束迈进大学校门开始，我们今后职业生涯的方向将逐步开始规划，无论当初你怎样想，你所填报的专业就是选择的起点。我们国家的高等教育越来越普及，每年的就业形势也不容乐观，理工类相关专业从事软件研发的较多，我身边的大多数学生选择这个专业方向主要是考虑以后好就业，不管喜欢不喜欢，反正跟着潮流走不会错。在这里我要特别提醒大家，专业选择是人生转折点，中国有句古语："男怕入错行，女怕嫁错郎"。学生的确应该冷静地分析自己的各科水平和能力，从兴趣爱好和价值观出发，要想清楚两个问题：一是所填报的专业和自己喜欢从事的职业是否一致？二是个人的理想与自身的学习能力是否一致？一个人只有干他愿意干的、认为有价值的、能力比较擅长的，以及个性禀赋适合的工作，才能更好地发挥潜能。我身边有一个现实的例子，就是我的一位同学，原来高中时他成绩非常好，基本是年级前三，但高考填志愿时只是按照老师的要求填报了一所重点大学的生物专业，多年后，据他自己说的确当年没有综合考虑自己的意愿和兴趣爱好，他根本就不喜欢生物，虽然他阶梯式地读到了博士，但最终他还是完全放弃了他的专业考到一个政府事业单位当了个完全与专业不相干的科干。最近遇到他，问起他的专业，他竟然做抓狂状说不要和他提生物，这位同学且不论他个人发展如何，这对于国家的教育资源就是一种浪费，培养一个学科博士国家需要投入多少人力物力，结果呢？在这里写这些话就是要强调选择专业是人生重要的转折点，要根据自己的具体情况具体分析，大胆而又慎重地填写好每一个专业的选择，用谨慎的

LAICAR.COM
shop35833438.taobao.com

态度和科学的方法真正做到从自身的情况出发，既然你选择了软件研发行业，就要喜爱它，尊重它，用心去对待它，才能信心百倍地迎接明天，在这个领域有长足的进步。

2 打下扎实基础，做好知识储备

本科阶段课程的学习是一切行业的基础，在软件研发行业尤为重要。最重要的两门基础课程就是数学和英语，业界有句话“数学水平决定你能走多远，英语水平决定你能走多快”，现在我是深有体会。李开复先生曾说过：“在全球范围内，美国的研究水平无疑是先进的。而除了美国之外，你会发现英国的研究水平也是相当突出的。究其原因，其实就是语言问题。英国人可以毫无阻碍地阅读美国乃至全球各种最新的英文研究报告和资料。这对于他们把握研究方向，跟踪最新进展，发表研究成果都有很大的帮助”。因此，英语学习对于我们做研发的人来说是相当重要的。只有加强这方面素质的培养，才能适应将来的发展。从业多年的朋友都知道英文原版教材更新速度远快于国内教程，且很多经翻译过的资料会失去原来精辟的讲解内涵，如果你英文水平足以阅读原版教材的话，会比别人进步快很多。工科专业必修的几门数学课程：高等数学、高等代数、概率论与数理统计、解析几何、计算方法、离散数学、初等数论等等必须是全部熟练掌握，最好精通（如果你想以后达到较高的设计层次）；英语与数学的学习可能会伴随你全部的职业生涯，很多在校学生对于这两类课程不是很重视，认为只是简单的基础了解，这非常不好。举个例子，现在很热门的3D类软件（游戏，绘图引擎等）设计会用到大量的积分思想与高等代数中的线性变换概念，而数据挖掘和预测决策分析类软件如果不懂数学分析与概率统计是根本没法做的，对于人工智能类软件的设计必须要求深入理解模糊数学理念。也许有朋友会说我是做数据库应用的，用不到什么高等数学类知识，很可惜你也错了，复杂库表结构组合的设计的合理与否完全要靠你对拓扑代数以及离散数学的理解程度，不要以为会几个SQL函数，会用IDE环境做出几个操作Oracle，DB2的应用就是会数据库开发了，这个仅仅是皮毛功夫而已，不理解数学概念的深层次应用，永远只能停留在初级水平，而且走不远的。举一个例子，一家国内比较大的公司设计，他们核心团队的研发经理对于数学知识的运用不是很好，典型的数据库类软件J2EE+WEBLOGIC+ORACLE的B/S架构，由于系统庞大，后端库表逻辑设计不合理，造成很多缺陷，最致命的是整体库结构与前端code不能分离，导致一个服务端口只能对应一个年度的库结构，做年终结转与年度初始化时痛苦万分。请多位专家综合分析之后，分析结果是核心架构必须推翻，80%的代码都要重构，我曾经寻问那个研发经理，做系统规划时考虑了代数拓扑的耦合度设计了吗？他一脸茫然，由此可见数学知识的重要性。

另外就是专业基础课的学习：计算机组成、操作系统原理、汇编语言、数据结构、编译原理、数据库原理、软件工程、结构性设计语言（Pascal、C）、面向对象设计语言（C++、C#）、计算机网络等，最好还深入学习一下算法分析、分布式系统、计算机图形学、形式逻辑、人工智能原理、软件设计模式、人机交互，软件构架/框架等的课程。在软件研发界，新技术层出不穷，但基本原理变化不大，我从业十多年来，越深入研究越体会到这些基础理论的重要性。比如我正在做的数学分析与D3D生物运动仿真引擎课题，就要用到大量的3D数学与专业基础理论，而且有些深层次没有理解透彻的概念还要回过头找这些资料学习参考。现在的大学普遍扩招，很多学生进入大学之后很茫然，对待课程学习也是纯应试目的，基本都是丢掉书本就忘了，很多教师在传授专业知识的时候也仅是浅尝辄止，不能形成知识的关联体系。计算机相关专业是实践性很强的学科，许多理论只看书是无法领会的，实践与理论必须相结合才行，这就需要我们自己做个有心人，现在的网络资源比十多年前要丰富多了，学习的时候可以多方面搜集相关资料，汇集理解，另外MIT的公开课很好，可以抽空看看，有助于加深理解。本科阶段的各项知识储备非常重要，我见到过很多计算机专业课学的很好的，但数学与英语基础不好，没办法继续深入研究下去。如果条件允许，读研是有必要的，可以加深专业方向的学习，提高职业素养与研究能力。其实最理想的软件研发学习之路是本科学数学，研究生读计算机相关专业，读博时再考虑在某个感兴趣的领域发展。如果有条件可以考虑一直深造下去，等到各项理论水平都很扎实了，再考虑投入工作。我见过一些所谓写代码的程序员，他们整天讨论自己掌握了多少编码技术，什么VB，Delphi，数据库玩得多顺等，其实他们忽略了软件研发最本质的东西，以至于多年之后仍然进步不大。现在有很多观点认为高等教育的重要性在下降，软件研发是吃青春饭的等等，其实这是极端错误的，在很多大公司的研究院里，高层次的研发专家基本都在35周岁向上，而一些研究院的首席软件科学家基本都是数学专业毕业，软件界的职业生涯有着典型的两类曲线。

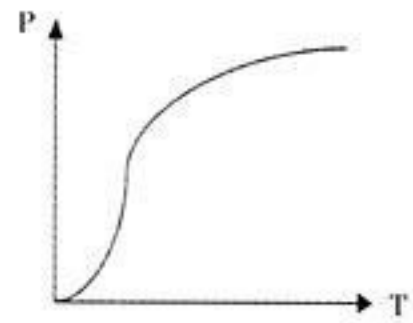


图1

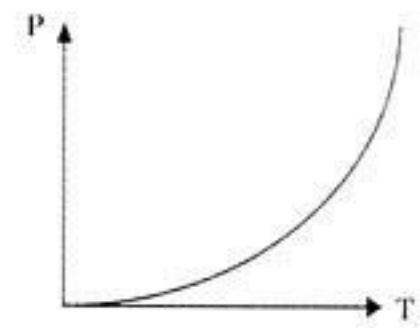


图2

这两幅图中T代表时间，P代表专业水平，很明显第一幅图中起步发展很快，但到了某一点时开始转折，发展会变慢而后将停留在某一极限而无法超越。第二幅图起步虽然很慢，但到了某一阶段量变转化为质变开始飞跃，而后无限向高层次发展。这两幅图就是著名的人月神话第一版所述的内容，已成为很多行业职业生涯的描述。写在这里是告诉大家，基础课程学习与知识储备的重要性，千万不要浪费青春，追求所谓快速入

来卡网出品
LAICAR.COM
shop35833438.taobao.com

门的短平快式的学习。说个很形象的例子：很多年轻人都喜欢看武侠小说，倚天屠龙记的张无忌为什么那么厉害，原因是他的内功基础非常扎实，学什么都很快，临场发挥现学现卖，超越了很多练了很多年的高手，其实这些知识储备就是你从业的内功，记得我从入门到运用 Java 以及 J2EE 设计一款 B/S 架构的组合算法模拟机只花了 2 个月时间，而且黑白盒测试一次性通过，这些都得益于我较扎实的知识储备。

3 注重理论学习的同时，选择合适的软件项目实践锻炼

无论你是在校学生，还是业界人士，理论学习的同时，配合实践锻炼是提升专业水平的必经之路。计算机相关专业是实践性很强的学科，所有的软件都是动手研发出来的。没听说过只看书，考试做题就能诞生软件专家。有个现实的例子：我以前有个同学很聪明，理论学习非常好，考试总是前几名，但他有个毛病，不喜欢静下心来动手搞研发，后来硕士毕业进入了一家很大的跨国公司核心研究机构，但由于总是不愿意动手，跟不上科研进度，最终不得不转行做了学院派，这是很可惜的。软件界的学习是 40%的理论，60%的实践，当然盲目的实践是没有效果的。现在各学校的教学资源都很好，计算机专业的学生获得实践锻炼的机会比以前多很多，找到一名理论与实践经验都很丰富的教授作为自己的第二老师是很有必要的，记得我后来就非常幸运地遇到了这样的导师，作为一名初学者可以尝试从普通的数据库类应用类软件开始动手，可以较为全面地接触一些数据库运用，网络通信及设计模式，编码规范等。特别要注意选择好自己使用的第一门开发语言工具，很多同学为了上手快喜欢用一些所谓的 RAD 类 IDE 环境，比如 Delphi，DOT NET 之类的，这些其实不利于良好编码习惯的养成，最严格规范的语言就是 C/C++系列，很多底层驱动与引擎都是基于 C/C++基础研发的，一个真正的研发专家必定是一个 C/C++语言专家。我当年独立设计的第一款 CMS 架构的物流仓储应用就是用 C++平台开发的，现在回想起来还是觉得很有益处。在校的学生尤其要注意理论学习与实践锻炼的同步。比如高等数学系列课程的学习可以结合 Matlab 的实验进行，有助于深入领会数学原理与算法学的基础。学习数据结构与算法分析学的时候有必要将教程的各个算法用语言逐个实现，对于汇编与计算机原理的学习可以尝试结合编写简单的显卡或声卡驱动去领会，操作系统的进程、线程池、多并发等机制一定不要只停留在基本原理的理解，动手做一个基础框架的内核模型会让你有很深的认识，学习操作系统时可以结合 Unix 设计原理与 Windows 内核编程等书籍同步学习，会让你对一些很抽象的概念有具体形象的认识。编译原理的学习可以加深对编译机制与人工智能初步的理解，可以尝试编写基本的编译框架。在校期间如果有合适的项目研发可以抽空参加，理论与实践的同步进行可以快速提升对专业的理解。当然，项目不是盲目去做的，各种类型的都应该尝试，数据库应用类是比较基础与简单的算法设计类、驱动程序、内核引擎等等都应该有所接触，对专业的各个领域应用都会有比较深的认识。尤其是毕业设计非常重要，一定要认真对待，选题尽量选择某个领域有一定深度的课题，我当年的本科毕业设计就是数据结构动态模拟课题，以 C++编码用计算机动画的方式将软件专业的 80 多个核心算法全部模拟出来，设计了一套一体化教学平台的软件，后来这款毕业设计软件被评为优秀设计并一直在学校作为教学案例保存。工作以后，我一直坚持理论学习与实践相结合，同时注意数学与英语学习的同步进行，为了将研发领域不断深入，后来进一步深造时利用空余时间历时 4 年系统学习了数学专业的全部课程，同时还在数学分析与概率统计及人工智能领域做了深入研究性学习，结合专业知识，在 10 余年间大小做了 90 多个项目，同时总结自己的实践经验写成各类论文并参加各类软件研发大赛，一步步从一线程序员走到现在的带队架构师，这都得益于不间断的理论学习与实践的相结合，希望我的一点学习经历对大家能有所帮助。

4 从小工到专家，在不断探索中积累渴望创造的激情，完成职业生涯进阶

经历了以上几个阶段的努力，基本上算可以进入职业软件研发的门槛了，但要记住你只能算是一个 Coder 小工而已，至于能走多远还要看自己的努力，在这里有几点经验供大家参考：

(1) 从编译原理的角度来理解工作中使用的高级语言。如果做到这一点，至少有两个好处：第一个好处是避免一大堆低水平重复出现的编译错误。一名优秀的 Coder 平均在一个工作日中应该完成 200 行以上的源码，其编译错误应该控制在 5 个以下，要知道这 200 行源码不是一次完成的，所以大多数情况下你都要追求一次编译通过，而一名职业水准的程序员，应该进一步做到即使 purify 这类的工具来检查源码，也不会存在严重的内存泄露；第二个好处是可以提高源码的可读性和效率。规范地编写你的代码使你自己的逻辑清晰，因为你明白多加几个括号和空行、多换行对齐、多注释，编译器是会自动识别的，不影响程序执行的效率，反过来，控制好递归调用和循环内的 if 语句才是提高程序效率的关键，要全力避免递归，但要深刻理解递归，能通过自己建立堆栈来把递归程序转换成非递归程序，要求还是较高的。

(2) 避免思维陷阱。人都有自己的思维惯性，这一定又会表现在你的程序逻辑中，有时你就是从这个惯性中跳不出来(谁都有这个时候)，但要心里有数才行，所以你需要帮助，如果你有几个水平相若或更高的职业伙伴，太好了，当遇到花 30 分钟还打不下的 bug 时，就别浪费时间了，找他们吧，最要紧的是能思路清晰明确地表述你的问题，通常你自己在这个过程中或者伙伴中就有人把问题解决了，又快又好。另外，有

LAICAR.COM
shop35833438.taobao.com

几个可以良性竞争的职业伙伴是人生的一件幸事，1+1>2，大家各有所长，你最好做到及时公开你的成果，技不压身，IT发展的这么快，你再优秀，那点东东也没有什么值得隐藏的，所以你可以技术或水平不够高，但千万不可以让真正具有职业水准的选手鄙视你的职业品质和行为。

（3）有自己debug的特点。这一点只是经验之谈，即使在VC这种高度完善的开放环境下，仍然应该要求自己仅凭打印语句就能debug。这也有两点好处：第一个好处是，遇到bug你会认真想问题所在，而不是用debug工具一步步简单地追踪卡在哪儿了，你定位bug范围的方式是从大到小、从粗到精，这是一种自顶向下的思维方式，而用工具追踪，容易形成自底向上的思维方式，这不算好，你应该先看到森林，再看到树木。程序就是逻辑过程，大多数程序从main函数开始，是由数据结构和功能子程序组成的一个树形结构的逻辑过程（要认清即使是面向对象的程序语言也是一样的），它的执行过程是深度优先的，但你定位bug应该是广度优先的；第二个好处是强迫你思考并记住而不是用工具看到调用过程，大脑的抽象逻辑思维能力和胳膊上肌肉的力量一样，都是练出来的，如果你的bug是程序结构上的逻辑错误引起的，这一点就非常重要了，顺便说一句，最难打的bug就是程序逻辑结构错误导致的bug。总之，程序员的职业水准：生产效率和程序质量，主要是取决于源码中bug的数量和debug的速度，而不是取决于编写源码的速度。给你一个我自己定义的考查一个职业程序员的指标：一个合格水准的职业程序员，编程的时间如果算一分的话，其累计debug的时间不能超过一分，真正职业高手累计debug的时间应该控制在0.5分以下，如何？你关上门悄悄问问自己，你花费在编程和debug上的时间比例是多少？如果你把程序员作为自己一生的职业，那么就永远都要牢记一点：追求做一个0 bug的优秀程序员！这是任何一个想成为职业程序员的人的理想，请相信：坚忍不拔地追求实现这个理想将让你出类拔萃！

（4）做好程序的单元测试。这是另一项考查你是否是一名具有合格职业水准的程序员的一个必要指标。其实在你拿到需求的时候就要准备单元测试用例了，并且这些用例将直接影响你的详细设计。举个例子，当你拿到一个需求时，除了分析它静态的功能外，还应明确它动态的操作/执行过程，把这个动态过程明确地用流程图画出来，比如分为A~Z的26步，其中A又可以进一步分解为A1~A5的5步，直到不能再分解为止。又比如说A3步不可分解了，那么你应该把A3步的正常操作和所有五花八门的异常操作都列出来，确保正常的操作肯定正确，异常的操作起码程序不退出才行。这样你就要写好多好多的测试用例，说句老实话，我也从来不写！但我一般会列一个提纲，比如A3步有正常的操作a、b、c、d、e共5项，异常的操作有f、g、h、i、j、k、l、m、n共9项，你在进行单元测试时都应该跑一遍，这样的程序都还不敢说质量如何好，但起码可以说较稳定吧！如果要想在进行单元测试时干得快、效率高，那么在进行详细设计时，你就应该把A3步中对所有正常操作和异常操作的判断都设计好，在编程实现A3步时，使得程序的结构合理高效。所以，如果你在工作中是割裂地看待软件工程中从需求、分析、设计、编码、测试等各个环节，恐怕水平很有限喔！但如果你在分析需求时就能看到测试的问题，并改进设计和实现，为此做好相应的准备工作，整个软件开发过程你的效率会高很多，通常你在一个开发团队中就会高度自信的，你已越过当一名偏颇、露骨的高手的境界，成为一个平静的高手，这可是The best in the best！因为别人看不出你高在哪儿，没见你有什么高招或特拼命干，但反正你就是干得又快又好、又省力。关于进行单元测试还有很多复杂的方法，这里只提到了最基本的一点，目的是让你在工作上考虑周全、安排有序，其他的还要靠自己慢慢领悟。

（5）要学会如何思考。如何处理工作及学习中的各项问题，因为思想决定行动，行动决定习惯，习惯决定命运。学习团队精神和沟通能力，培养团队合作的能力和领导才能，也可以发挥你的专业特长。但更重要的是，你要做一个诚心诚意的服务者和志愿者，或扮演沟通桥梁的角色，并以此锻炼自己的沟通能力。把握不用“付学费”的学习人际交往的机会，犯了错误也可以从头来过。一个有为的公司，员工比拼的是胸怀。一流的人才招聘的是一流的人才，二流的人才招聘的是三流的人才。要与优秀的员工做好朋友，做你一生的朋友，和优秀的人接触会改变你的一生。只有胸怀广阔的人才可以有所作为，否则就是平庸。只有这样才可以成为一位合格的软件人员。

（6）不要只关注专业相关的理工科类知识。人文、历史、哲学有空也可以看看，对启发创造力很有帮助，日常工作中你采用何种设计架构模式都是具有创造性的，创造力的培养不单单是专业知识的范畴，有很多人文修养在其中。要竖立正确的人生观，价值观，坚定自己的专业方向，有条件一定要继续读研深造。需要说明的是，选择研发的人未必个个都是高手，严格来说，大部分都不会编程序。也就是说，庸庸碌碌之辈仍然占绝大多数。研究生毕业的师兄只拿3千左右的比比皆是，所以不要寄希望于拿一张研究生文凭出去赚高薪。在继续深造前要想想自己目前具备的能力，而不是混文凭，以下供参考：

1）你已经认为C++和汇编语言都是很简单的语言，并能够自如地运用；

2）你能够在30分钟之内想到正确的五子棋AI算法设计思路和方向；

3）你完全理解STL为什么这么重要；

4）你能够独立地解决所有的编译与链接问题，哪怕你从来没有遇到的问题，你也不需要询问任何人；

5）英文网站是你的首要信息来源；

6）能够读懂英语写成的国际标准，比如NTFS磁盘格式标准；

来卡网出品 LAICAR.COM shop35833438.taobao.com

7）你经常站在集合论的角度思考算法问题；

8）能够理解一个简单的驱动程序，能够理解一个简单 3D 交互程序；

9）你能够认识到线性代数和概率论在实际编程工作中的极端重要性；

10）你完全理解 COM 的设计思想，尤其能够理解 COM 为什么要设计成这样；

11）当我说到虚函数的重要作用时，你不会急着去找书来翻；

12）你能够说出 C++为什么比其他语言优秀的理由。

如果你同时具备 5 条以上，可以认为你已经具备较好的开发经验了。在这种状态下深造读研，你将取得最大的效益值。

从小工到专家，在不断探索中积累渴望创造的激情，完成职业生涯进阶，特别注意进阶深造期间，你一定要做有理论深度的算法设计，比如大规模数据的搜索算法，性能是首要考虑因素，不要奢望 SQL 函数能够帮你解决问题，所有的问题你都必须自己解决，你必须解决内外存交换的性能瓶颈。再比如极品飞车的 3D 场景生成、图形变换、碰撞检测、物性模拟、纹理映射、灯光模型等等，这些都是必须精通的技术。要是你认为深造多年之后还是要去搞一般的程序设计，如信息管理系统之类的软件，那么你职业生涯的价值就完全不会得到充分的体现。

5 选择合适的领域，自主创新核心技术，为振兴国产软件出力

以正确的学习方法经过坚持不懈的努力，你终将在软件研发的某一个领域有所成就，可能你已成为某个领域的专家，这时你可以专注地在某个领域发展，而不必担心基本的衣食住行问题。这时如果你有心，建议你考虑支持一下我们民族软件的发展，考虑一下自主创新一些核心的技术。

在此借机呼吁一下，我们广大的 IT 人士，各类信息化公司，多投些精力做一下系统级的基础研发，有能力多关注一下自主创新核心技术，让我们自主研发的操作系统、编译器、数据库、D3D 引擎等等基础核心技术的平台能够在祖国大地生根发芽，结出累累硕果。有句话“师夷长技以制夷”，还有句话“天下兴亡匹夫有责”，仅做为一名从业十多年的老鸟在这里写下这些话，期盼广大的正在入行及一起奋战的 IT 同仁们，今天的初学者，也是明天的领航人，让我们一起为振兴国产软件努力奋斗。祝愿祖国更加强大，明天更美好！

一种控制屏幕光标的方法

该方法阐述了一种利用手机或平板电脑（以下统称为手机）的移动来操作本身屏幕光标或其它设备屏幕光标的方法，该方法的核心思想是将手机或平板电脑本身作为一个鼠标来控制本身屏幕光标或其它设备屏幕光标的移动。

权利要求一：利用手机的移动来操作本身屏幕光标或其它设备屏幕光标，该移动可以是在某一固定平面移动也可以是三维立体空间移动。

权利要求二：当利用手机或平板电脑的移动作为其它设备的鼠标时，还可以将手机作为一种身份认证设备，即没有对应的手机，不能操作或登录电脑。

该方法的实现：

(1) 利用手的移动来操作本身屏幕光标

◆将手机作为一个光电鼠标来看待：这里可以利用手机本身的摄像头或加装特殊的光电装置（以下统称为摄像头）来判断手机是否移动及移动的距离。手机的操作系统或应用程序分别在前后的时间从摄像头中获取得两幅图像，然后判断两幅图像的特征点的位置是否发生了变化及变化的距离，从而根据两幅图像的位移来控制屏幕光标的移动，其原理与光电鼠标类似，这里就不再详述。

◆将手机作为一个机械鼠标来看待：这里可以在手机上底部或侧部加滚动装置来判断手机是否移动及移动的距离，手机的操作系统或应用程序根据滚动装置的位移来控制屏幕光标的移动，其原理与机械鼠标类似，这里就不再详述。

(2) 利用手机的移动来操作其它设备（例如个人电脑，以下统称为电脑）的屏幕光标

◆将手机作为一个无线光电鼠标来看待：手机与电脑的通讯方法可以是手机本身的蓝牙与电脑的蓝牙或其它的无线通讯方式。手机通过摄像头来判断手机是否移动及移动的距离，然后将其本身的位移及相应控制键的变化发送给电脑，电脑根据该信息来控制电脑屏幕光标的移动或击键。

◆将手机作为一个无线机械鼠标来看待：手机与电脑的通讯方法可以是手机本身的蓝牙与电脑的蓝牙或其它的无线通讯方式。手机通过滚动装置来判断手机是否移动及移动的距离，然后将其本身的位移及相应控制键的变化发送给电脑，电脑根据该信息来控制电脑屏幕光标的移动或击键。

◆手机作为鼠标的输入设备时，还可以作为一种身份认证设备，即没有对应的手机，不能操作或登录电脑，手机与电脑的认证方式，可以是存中手机中的硬件唯一编码或自定义编码。

为了防止光标占据屏幕，当手机在一定时间内没有位移时，自动隐藏光标，当发生移动时再出现光标。

LAICAR.COM
shop35833438.taobao.com

利用语义网技术实现铁路交通的地理语义查询（二）

——从关系数据库中创建本体与定义推理规则

董 志

摘 要：介绍了根据 SQLite 关系数据库中的数据创建火车站点本体并生成 OWL 文件，分析了铁路交通的模糊查询和精确查询，定义相应的本体语义推理规则和 SPARQL 语句，使用 Jena 的推理机设计本体推理类，这一部分是实现铁路交通地理语义查询的核心工作。

关键词：语义网；Web3.0 标准；本体构建；Jena 工具；地理本体；逻辑推理；SPARQL 语句

1 火车站点的本体设计

在上一篇中，程序完成了基于 Jena 的创建本体的类-CreateOwlByJena。在本篇中，程序利用该类与 SQLite 数据操作类 SQLiteDB，将数据表中的数据自动生成 RDF 格式的本体并创建本体文件。

本体 (ontology) 是一个哲学上的概念，是指哲学中研究世界的本原或本性的部分。从本体论的基本概念出发，哲学上把本体定义为“对世界上客观事物所进行的系统描述”。本体在计算机领域是指定义元数据及其相关关系的规范。它依赖于所采用的语言，按照表示和描述的形式化程度不同，可以分为完全非形式化、半形式化和严格形式化。本体的形式化程度越高，越有利于计算机进行自动处理。本体最为突出的应用是智能信息检索。而本文介绍的程序中使用的本体是使用 RDF 表示的。

RDF (资源描述框架) 是描述资源的一项标准 (W3C 的推荐标准)，可以把 RDF 考虑成由节点和箭头的形式构成的图。一个简单的 capitalOfprovince RDF 节点属性在 RDF 中可能看起来如图 1 所示。

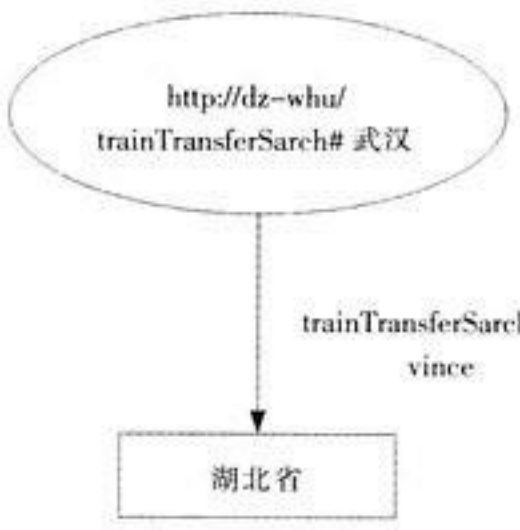


图 1 capitalOfprovince RDF 结点显示效果

本体头部一般为：

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
   xmlns:rdf = "http://www.w3.org/1999/02/22 -rdf -syntax -ns#"
   xmlns:owl="http://www.w3.org/2002/07/owl#"
                xmlns:trainTransferSarch = "http://dz -whu/trainTransferSarch#"
   xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >
 ...
```

简单的类定义：

```
            <rdf:Description  rdf:about = "http://dz -whu/trainTransferSarch#trainStop">
     <rdf:type  rdf:resource = "http://www.w3.org/2002/07/owl#Class"/>
  </rdf:Description>
  ...
```

简单的属性定义：

```
            <rdf:Description  rdf:about = "http://dz -whu/trainTransferSarch#startTrainStop">
     <rdf:type  rdf:resource = "http://www.w3.org/2002/07/owl#ObjectProperty"/>
  </rdf:Description>
...
```

简单的个体定义：

```
            <rdf :Description  rdf:about = "http://dz -whu/trainTransferSarch# 北京">
                  <rdf:type  rdf:resource = "http://dz -whu/trainTransferSarch#capitalOfprovince"/>
    <trainTransferSarch:ID>27</trainTransferSarch:ID>
    <trainTransferSarch:province>北京市</trainTransferSarch:province>
     <trainTransferSarch:shortName > 京 </trainTransferSarch:shortName>
    <trainTransferSarch:cityName > 北 京 </trainTransferSarch:cityName>
  </rdf:Description>
   ...
            <rdf:Description  rdf:about = "http://dz -whu/
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

FOLLOW MASTER PROGRAM

```
trainTransferSarch# 北京南-济南">
    <trainTransferSarch:cid>D341</trainTransferSarch:cid>
    <trainTransferSarch:cid>D333</trainTransferSarch:cid>
                        <rdf:type rdf:resource = "http://dz -whu/
trainTransferSarch#trainStop"/>
    <trainTransferSarch:qid>北京南</trainTransferSarch:qid>
    <trainTransferSarch:zid>济南</trainTransferSarch:zid>
     <trainTransferSarch:qLocation>
     http://dz-whu/trainTransferSarch# 北京
     </trainTransferSarch:qLocation>
     <trainTransferSarch:zLocation>
     http://dz-whu/trainTransferSarch# 济南
     </trainTransferSarch:zLocation>
  </rdf:Description>
```

2 根据 SQLite 表中数据生成火车站点本体

该部分程序的功能是从 SQLite 关系数据库中抽取记录，根据表名、表结构、记录创建本体中的类、属性与个体。将关系数据库中数据转换为以上 RDF 本体的实现代码如下：

```
...
using com.hp.hpl.jena.db;
using com.hp.hpl.jena.rdf.model;
using com.hp.hpl.jena.ontology;
using System.Collections;
using jenaDealTrainStop;
...
class RelationDbToOwl
  {
      private OntModel _destinationowl = ModelFactory.
createOntologyModel();
      private static String _ns = "trainTransferSarch";
       private static String _baseURI = "http://dz -whu/
trainTransferSarch#";
      public OntModel destinationowl
      {
         get { return _destinationowl; }
         set { _destinationowl = value; }
      }
      public static string ns
      {
         get { return _ns; }
         set { _ns = value; }
      }
      public static string baseURI
      {
         get { return _baseURI; }
         set { _baseURI = value; }
      }
     //创建规划时需要的本体属性
      private OntModel createOtherOwl()
      {
                OntModel model = _destinationowl;
                    CreateOwlByjena createModelOwl = new
CreateOwlByjena(model);
                String ns = _ns;
                createModelOwl.setOwlPreFix(ns, baseURI);
                ObjectProperty owlProperty = (ObjectProperty)
createModelOwl.createOwlProperty("transferType", "0");
                        OntClass ontClass = model.getOntClass
(_baseURI + "trainStop");
                owlProperty.addDomain(ontClass);
                owlProperty.addRange(model.getOntClass("http:
//www.w3.org/2002/07/owl#DatatypeProperty"));
                            createModelOwl.createOwlProperty ("
midTrainStop1", "0");
                            createModelOwl.createOwlProperty ("
midTrainStop2", "0");
                            createModelOwl.createOwlProperty ("
startTrainStop", "0");
                            createModelOwl.createOwlProperty ("
endTrainStop", "0");
                            createModelOwl.createOwlProperty ("
hasReach", "0");
                return model;
            }
      //根据数据表中表名、表结构、记录创建本体类、属性、个体
            private OntModel dbConvertOwl (DataSet
sourceRs, string TableName,string IndividualNameFields)
            {
                OntModel model = _destinationowl;
                 CreateOwlByjena createModelOwl = new
CreateOwlByjena(model);
                String ns = _ns;
                createModelOwl.setOwlPreFix(ns,baseURI);
                            string [] IndividualNameTemp
= IndividualNameFields.Split(',');
                for (int i = 0; i < sourceRs.Tables.Count; i++)
                {
                   ArrayList DataColumnAL = new ArrayList();
                   string ConstraintsFileds = "";
      SQLiteDB sqliteDbObj = new SQLiteDB();
                   sqliteDbObj.openConnection("");
                    DataTable foreignKeysDT = sqliteDbObj.
getRSSchema("ForeignKeys");
                foreach (DataRow dr in foreignKeysDT.Rows)
                    {
                        ConstraintsFileds += "," + dr ["
FKEY_FROM_COLUMN"].ToString();
                    }
                    foreach (DataColumn column in sourceRs.
Tables[i].Columns)
                    {
                        if (ConstraintsFileds.IndexOf (column.
```


LAICAR.COM
shop35833438.taobao.com

```
ColumnName) >= 0)
                {
                    DataColumnAL.Add(0);
                }
                  if    (ConstraintsFileds.IndexOf  (column.
ColumnName) < 0)
                {
          DataColumnAL.Add(1);
                }
            DataColumnAL.Add(column.ColumnName);
              }
                          createModelOwl.createOwlClass
(TableName, DataColumnAL);
                          OntClass tableOntClass = model.
getOntClass(_baseURI + TableName);
                for  (int j = 0; j < sourceRs.Tables[i].Rows.
Count; j++)
                {
                   string IndividualName = "";
                    foreach  (string IndividualFilesName in
IndividualNameTemp)
                   {
                      if (IndividualName.Length > 0)
                      {
                         IndividualName += "-" + sourceRs.
Tables[i].Rows[j][IndividualFilesName].ToString();
                      }
                      else
                      {
                          IndividualName = sourceRs.Tables
[i].Rows[j][IndividualFilesName].ToString();
                      }
                   }
                    Individual rsIndividual = createModelOwl.
createOwlIndividual(_baseURI + IndividualName, tableOnt
Class);
                   for  (int k = 0 ; k < sourceRs.Tables[i].
Columns.Count;k++)
                   {
                       int index = DataColumnAL.IndexOf
((object)sourceRs.Tables[i].Columns[k].ColumnName);
                if (DataColumnAL[index - 1].ToString() == "0")
                                          createModelOwl.
setOwlIndividualObjectPropertyValue (_baseURI + sourceRs.
Tables [i].Columns [k].ColumnName, _baseURI + sourceRs.
Tables[i].Rows[j][k].ToString(), rsIndividual);
                if (DataColumnAL[index - 1].ToString() == "1")
      createModelOwl.setOwlIndividualDatatypePropertyValue
(_baseURI + sourceRs.Tables [i].Columns [k].ColumnName,
sourceRs.Tables[i].Rows[j][k].ToString(), rsIndividual);
                       }
                }
                }

                return model;
            }
            //更新火车站点位置
                    private void dealTrainStopByLocationCity
(SQLiteDB sqliteDbObj)
            {
                string sqlStr = "";
                        sqliteDbObj.ExecuteSQLiteSQL("update
trainStopLocationCity set weight = 1 where trainStopName
= cityName");
                        sqliteDbObj.ExecuteSQLiteSQL ("update
trainStop set qLocation = '',zLocation = ''");
                DataSet db = sqliteDbObj.getRSFromDb("select
trainStopName,cityName from trainStopLocationCity");
                for (int i = 0; i < db.Tables[0].Rows.Count; i++)
                {
                    sqlStr = "update trainStop set  qLocation = '"
+ db.Tables[0].Rows[i]["cityName"] + "' where  '" + db.Tables
[0].Rows[i]["trainStopName"].ToString().Replace("火车","") + "'
like '%'||qid ";
                    sqliteDbObj.ExecuteSQLiteSQL(sqlStr);
                    sqlStr = "update trainStop set  zLocation = '"
+ db.Tables[0].Rows[i]["cityName"] + "' where  '" + db.Tables
[0].Rows[i]["trainStopName"].ToString().Replace("火车", "") + "'
like '%'||zid ";
                        sqliteDbObj.ExecuteSQLiteSQL(sqlStr);
                    }
                }
                public void createOntology()
                {
                    SQLiteDB sqliteDbObj = new SQLiteDB();
                    sqliteDbObj.openConnection("");
                    dealTrainStopByLocationCity(sqliteDbObj);
                    //创建 capitalOfprovince 本体和 trainStop 本体
                    DataSet db = sqliteDbObj.getRSFromDb("select
id,province,shortName, cityName from capitalOfprovince");
                     dbConvertOwl  (db,  "capitalOfprovince",  "
cityName");
                    db.Clear();
                    db = sqliteDbObj.getRSFromDb("select qid,zid,
qlocation,zlocation,cid from trainStop");
                    dbConvertOwl(db,"trainStop","qid,zid");
                    createOtherOwl();
                    //利用 java.io 写本体文件
                java.io.File fs = new java.io.File("data\\trainStop.owl");
                    if (! fs.exists())
                    {
                        fs.createNewFile();//不存在则创建
                    }
                    java.io.OutputStreamWriter os = null;
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
            os = new java.io.OutputStreamWriter(new java.
io.FileOutputStream("data\\trainStop.owl"), "utf-8");
            _destinationowl.write(os);
        }
    }
```

创建的一部分本体在 Protégé 的 OntoGraf 页面中显示结果如图 2 所示。

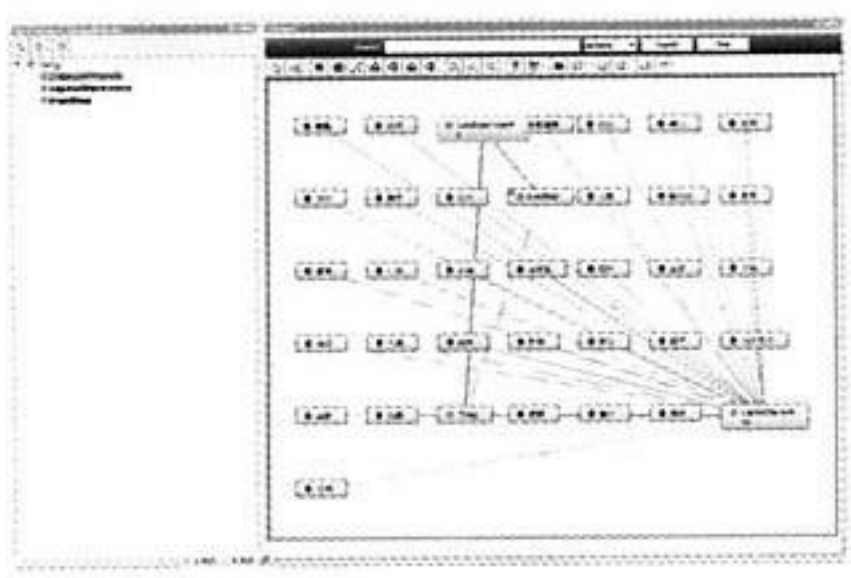


图 2 capitalOfprovince 类及个体在 OntoGraf 页面中显示结果

3 Jena 的规则推理与 SPARQL 查询本体

给定了本体和模型后，Jena 的推理引擎可以派生模型未明确表达的其他语句。Jena 提供了多个 Reasoner 类型来使用不同类型的本体。Jena 的推理是在规则的指导下进行的。规则是语义网中表示知识的一种方式，它通常超过了 OWL 的表达能力，语义 Web 上的规则是典型的条件语句：if-then 字句。只有当特定的陈述为真，才会添加新的知识。本体规则是对本体逻辑描述的必要补充。定义规则通过语义推理，本体能够构建潜在的知识，如：a 到达 b，b 到达 c，则 a 可到达 c。

Jena 提供了一个基于规则的推理机用来实现 RDFS 和 OWL 推理机，Jena 框架自带了推理引擎和所识别的规则语言，它也可以作为一般使用。推理机提供了对 RDF 图基于规则的推理，还提供了 3 种模式，分别为 forward chaining，backward chaining 和 hybrid。其推理机制结构如图 3 所示。

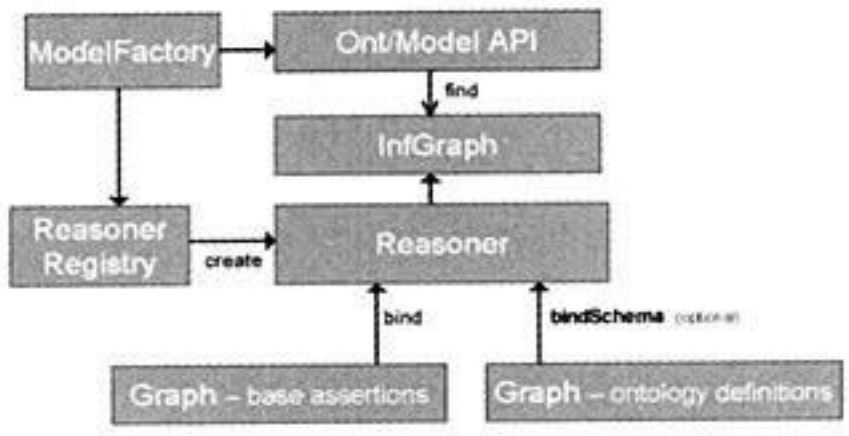


图 3 Jena 推理机制结构图

一个 Rule 是由 Jena 的 Rule 对象定义，包含：前提，结论，一个可选的名字和一个可选的方向。一个规则集合就是一个 Rules 的 List。一个简单的 parser 被包含在 Rule 里，它允许 rules 在文本源文件中以紧凑的方式指定。也可以创建自己的 rule parser。

一个非正式的简化的文本 rule 语法的描述如下：

```
Rule:=bare-rule .
        or  [ bare-rule ]
          or  [ ruleName : bare-rule ]
bare-rule :=   term, ... term -> hterm, ... hterm
// forward rule
  or  term, ... term <- term, ... term    // backward rule
hterm :=   term
        or  [ bare-rule ]
term :=   (node, node, node)          // triple pattern
        or  (node, node, functor)  // extended triple pattern
    or  builtin(node, ... node)  // invoke procedural primitive
functor :=   functorName(node, ... node)  // structured literal
node :=   uri-ref              // e.g. http://foo.com/eg
        or  prefix:localname                  // e.g. rdf:type
        or  <uri-ref>              // e.g. <myscheme:myuri>
        or  ?varname                  // variable
        or  'a literal'                // a plain string literal
        or  'lex'^^typeURI
// a typed literal, xsd:* type names supported
        or  number                          // e.g. 42 or 25.5
```

程序在进行火车到达推理时按照以上语法，以“bare-rule”的向后模式，规则应写为：

（1）模糊检索：根据出发地到达目的地语义查询：

1）直接到达：[rule1: (?a base:qLocation 起始站城市) (?a base:zLocation 终点站城市) -> (?a base:到达 ?b)]。

2）一次换乘到达：[rule2: (?a base:zid ?midTrainStop1) (?b base:qid ?midTrainStop1) (?a base:qLocation 起始站城市) (?b base:zLocation 终点站城市) -> (?a base:到达 ?b) (?a base:中间站点是 ?midTrainStop1)]。

（2）精确检索：根据火车站点名称查询。

1）直接到达：[rule1: (?a base:qid 起始站名称) (?a base:id 终点站名称) (?a base:zid ?zid) (?a base:qid ?qid) -> (?a base:到达 ?b)]

2）一次换乘到达：[rule2: (?a base:zid ?midTrainStop1) (?b base:qid ?midTrainStop1) (?a base:qLocation 起始站城市) (?b base:zLocation 终点站名称) -> (?a base:到达 ?b) (?a base:中间站点是 ?midTrainStop1)]。

（3）混合检索：根据精确或者模糊查询。

1）起点模糊、终点精确直接到达：[rule1: (?a base:qLocation 起始站城市) (?a base:zid 终点站) -> (?a base:到达 ?b)]。

2）起点模糊、终点精确一次换乘到达：[rule2: (?a base:zid ?midTrainStop1) (?b base:qid ?midTrainStop1) (?a base:qLocation 起始站城市) (?b base:zid 终点站) -> (?a

LAICAR.COM
shop35833438.taobao.com

base:到达 ?b)　(?a base:中间站点是 ?midTrainStop1)]。

3）起点精确、终点模糊直接到达：[rule1: (?a base:qid 起始站)　(?a base:zLocation 终点站城市) -> (?a base:到达 ?b)]。

4）起点精确、终点模糊一次换乘到达：[rule2: (?a base:zid ?midTrainStop1)　(?b base:qid ?midTrainStop1)　(?a base:qid 起始站)　(?b base:zLocation 终点站城市) -> (?a base:到达 ?b)　(?a base:中间站点是 ?midTrainStop1)]。

建立以上规则后，推理机对象在运行一次后即可将包括模糊和精确检索的以上规则推导完成。通过这一点可以看出，本体推理查询比关系数据库 SQL 查询要方便许多。

在本体进行推理后，本体会在原三元组结构及定义进行调整，产生新的本体三元组定义。推理完毕后的下一步往往需要将推理结果检索出来，这就需要使用到 SPARQL。SPARQL 是 W3C 的 RDF 数据工作组设计的一种查询语言和协议，用于 RDF 数据的查询。基于图模式匹配的，最简单的图模式是三元组模式，它是像一个 RDF 三元组，但在任何主语、谓语或宾语的位置中可能有变量，如：{?trainStop base: midTrainStop1 ?midTrainStop}。SPARQL 语法结构和 SQL 语法比较相近。SPARQL 查询变量有全球范围；查询时，同一个名字在各处是相同的变量。变量用“?”指出；“?”不构成变量的组成部份。“$”是“?”的替代品。在查询中，$abc 和?abc 是相同的变量。三元组模式被写作为一列主语、谓语、宾语；有简单的方法来写一些通用的三元组模式构造。三元组模式用 {} 被聚集在一起。

使用 SPARQL 语句查询规则推理结果如下：

```
select ?reachType ?startTrainStop ?midTrainStop1 ?midTrainStop2 ?endTrainStop where {
        ?a base:hasReach ?reachType .
        ?a base:midTrainStop1 ?midTrainStop1 .
        ?a base:midTrainStop2 ?midTrainStop2 .
        ?a base:startTrainStop ?startTrainStop .
        ?a base:endTrainStop ?endTrainStop .
        } order by ?reachType
```

语句执行后，得到事例结果如图 4 所示。



图 4　SPARQL 检索结果

在本程序中的推导与查询主要用到了 com.hp.hpl.jena.query、com.hp.hpl.jena.reasoner、com.hp.hpl.jena.reasoner.rulesys 三类命名空间中的对象，先通过 Rule 对象编译规则，然后使用 Reasoner 进行规则推理，最后使用 Query 对象检索得到推理结果，其具体代码实现如下：

```
...
using jena;
using com.hp.hpl.jena.db;
using com.hp.hpl.jena.rdf.model;
using com.hp.hpl.jena.query;
using ikvm.lang;
using com.microsoft.sqlserver.jdbc;
using System.Xml;  //在.net 中操作 OWL 或 RDF 还是要
//引入 xml 库
using System.Xml.XPath;
using ikvm.io;
using com.hp.hpl.jena.ontology;
using com.hp.hpl.jena.reasoner;
using com.hp.hpl.jena.reasoner.rulesys;
using com.hp.hpl.jena.vocabulary;
using System.Collections;
...
public class JenaRule
  {
 OwlPreFix _defaultOwlPreFix = null;
      string _reasonResultStr = "";
      public OwlPreFix defaultOwlPreFix
      {
          get { return _defaultOwlPreFix; }
          set { _defaultOwlPreFix = value; }
      }
      public string reasonResultStr
      {
          get { return _reasonResultStr; }
          set { _reasonResultStr = value; }
      }
      /**

       * jena 原生推理机推理

       */
       public string reasoner (String ruleString, String filePathString)
       {
                  OntModel ontModel = ModelFactory.createOntologyModel();
            ontModel.read(filePathString);
                        Reasoner reasoner = new GenericRuleReasoner(Rule.parseRules(ruleString));
                        InfModel resulInfModel = ModelFactory.createInfModel(reasoner, ontModel);
      StringBuilder querystringBuffer = new StringBuilder();
            querystringBuffer.Append("PREFIX owl: <http://www.w3.org/2002/07/owl# > PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> ");
            querystringBuffer.Append ("PREFIX base:<" + _defaultOwlPreFix.preFixStr + "> ");
            querystringBuffer.Append("select ?reachType ?
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

FOLLOW MASTER PROGRAM

```
startTrainStop ?midTrainStop1 ?midTrainStop2  ?endTrainStop
where { "
                + "?a  base:hasReach  ?reachType . "
          + "?a  base:midTrainStop1  ?midTrainStop1 . "
          + "?a  base:midTrainStop2  ?midTrainStop2 . "
                + "?a  base:startTrainStop  ?startTrainStop . "
                + "?a  base:endTrainStop  ?endTrainStop . "
                + "} order by ?reachType");
                        Query query = QueryFactory.create
(querystringBuffer.ToString());
                        QueryExecution queryExecution =
QueryExecutionFactory.create(query, resulInfModel);
          return  com.hp.hpl.jena.query.ResultSetFormatter.
asXMLString(queryExecution.execSelect());
     }
          //Rule 规则定义
                        public string JenaRuleReasoner (string
startCityName, string endCityName)
          {
             string pathFile = "file:data\\trainStop.owl";
             JenaRule jenaRule = new JenaRule();
                        StringBuilder ruleStringBuffer = new
StringBuilder();
               ruleStringBuffer.Append  (" [rule1: (?a  base:
qLocation '" + _defaultOwlPreFix.preFixStr + startCityName
+ "')  (?a base:zLocation '" + _defaultOwlPreFix.preFixStr +
endCityName + "') (?a base:zid ?zid)  (?a base:qid ?qid)-> (?a
base:midTrainStop1 ?')  (?a base:midTrainStop2 ?')  (?a base:
hasReach  0)  (?a  base:endTrainStop  ?zid)  (?a  base:
startTrainStop ?qid)]");
               ruleStringBuffer.Append("[rule2: (?a  base:zid  ?
midTrainStop1)  (?b  base:qid  ?midTrainStop1) (?a  base:
qLocation '" + _defaultOwlPreFix.preFixStr + startCityName
+ "')  (?b base:zLocation '" + _defaultOwlPreFix.preFixStr +
endCityName + "') (?a base:qid ?qid)  (?b base:zid ?zid)-> (?
a base:midTrainStop1 ?midTrainStop1)  (?a base:midTrain
Stop2 ?')(?a base:hasReach 1) (?a base:startTrainStop ?qid) (?
a base:endTrainStop ?zid)  ]");
               ruleStringBuffer.Append("[rule3:(?a base:qid '" +
startCityName + "')  (?a base:zid '" + endCityName + "') (?a
base:zid ?zid)  (?a base:qid ?qid)->  (?a base:midTrainStop1
?') (?a base:midTrainStop2 ?') (?a base:hasReach 0) (?a base:
endTrainStop ?zid)  (?a base:startTrainStop ?qid)]");
               ruleStringBuffer.Append("[rule4: (?a  base:zid  ?
midTrainStop1)  (?b  base:qid  ?midTrainStop1)  (?a base:qid
'" + startCityName + "')  (?b base:zid '" + endCityName + "')
(?a  base:qid  ?qid)  (?b  base:zid  ?zid) ->  (?a  base:
midTrainStop1 ?midTrainStop1)  (?a base:midTrainStop2 ?')  (?
a base:hasReach 1)  (?a base:startTrainStop ?qid)  (?a base:
endTrainStop ?zid)  ]");
               ruleStringBuffer.Append  (" [rule5: (?a  base:
qLocation '" + _defaultOwlPreFix.preFixStr + startCityName
+ "') (?a base:zid '" + endCityName + "') (?a base:zid ?zid)  (?
a base:qid ?qid) ->  (?a base:midTrainStop1 ')  (?a base:
midTrainStop2  ')  (?a base:hasReach  0)  (?a  base:
endTrainStop ?zid)  (?a base:startTrainStop ?qid)]");
               ruleStringBuffer.Append("[rule6: (?a  base:zid  ?
midTrainStop1)  (?b  base:qid  ?midTrainStop1)  (?a base:
qLocation '" + _defaultOwlPreFix.preFixStr + startCityName
+ "')  (?b base:zid '" +endCityName + "')  (?a base:qid ?qid)
(?b base:zid ?zid)-> (?a base:midTrainStop1 ?midTrainStop1)
(?a base:midTrainStop2 ')  (?a base:hasReach 1)  (?a base:
startTrainStop ?qid) (?a base:endTrainStop ?zid)  ]");
               ruleStringBuffer.Append("[rule7:(?a base:qid '" +
startCityName  +  "')  (?a  base:zLocation  '" +
_defaultOwlPreFix.preFixStr + endCityName + "') (?a base:zid
?zid)  (?a base:qid ?qid) ->  (?a base:midTrainStop1 ') (?a
base:midTrainStop 2 ')  (?a base:hasReach 0)  (?a base:
endTrainStop ?zid)  (?a base:startTrainStop ?qid)]");
               ruleStringBuffer.Append("[rule8: (?a  base:zid  ?
midTrainStop1)  (?b  base:qid  ?midTrainStop1)  (?a base:qid
'" + startCityName + "')  (?b base:zLocation '" +
_defaultOwlPreFix.preFixStr + endCityName + "')  (?a base:
qid ?qid)  (?b base:zid ?zid) ->  (?a base:midTrainStop1 ?
midTrainStop1)  (?a  base:midTrainStop2  ')  (?a  base:
hasReach 1)  (?a base:startTrainStop ?qid)  (?a base:
endTrainStop ?zid)  ]");
                com.h  p.hpl.jena.util.PrintUtil.registerPrefix ("
base", _defaultOwlPreFix.preFixStr);
               com.hp.hpl.jena.util.PrintUtil.registerPrefix("owl",
"http://www.w3.org/2002/07/owl#");
                _reasonResultStr =reasoner (ruleStringBuffer.
ToString(), pathFile);
               return _reasonResultStr;
          }
     }
```

需要注意的是：Jena 在内存中直接读入规则会有问题（读入规则文件是没有问题的），通过研究 Jena 推理类源程序可以得出其产生的原因是命令空间没有定义，因此如果需要直接添加 rule，则程序必须在推理前用 PrintUtil.registerPrefix 方法注册命名空间的名称。

4　规则推理结果

当查询“XX-武汉”时，程序将会自动推导出以下模糊结果：在武汉市内的“武昌”、“汉口”、“武汉”3 个火车站被检索到，如图 5 所示。

通过其他数据进行测试，推理机程序成功地将各种查询组合方式都进行了处理。无论按照精确或者模糊查询，最后根据规则仍能推导出结果，当精确查询“XX—余杭”，查询结果如图 6 所示。

（下转第 26 页）

LAICAR.COM
shop35833438.taobao.com

不会编译原理也能做 Android 计算器

李 宁

摘 要：通常比较复杂的计算器都需要编译原理的知识，例如，需要计算表达式的值。现在已有很多可以自动生成计算规则的工具（如 JavaCC），可以不需要了解编译原理的知识就可以实现非常复杂的计算器程序。结合 JavaCC 自动生成可以解析表达式的 Java 代码，并编写 Android 版的计算器程序。

关键词：Android 平台；计算器；JavaCC 工具；编译原理；表达式

1 引言

也许有很多人看到“编译器”3 个字，总会想起大学中最恐怖的课程“编译原理”。当然，也可能会想到目前一些主流的编程语言（如 Java、C#、C++等）。这些语言都有各自的编译器。我们接触的这些语言主要是使用它们的编译器，一般并不涉及到底层的编译技术。然而，编译技术不仅仅只被使用在这些编程语言中，在开发过程中，编辑技术几乎无处不在。例如，解析 XML 文档，分析某些带结构的文本，计算一个表达式，这些都会使用到编译技术。

可能有的人认为，实现编译器或其他类似的产品需要很高的技术水平，只有大师级的程序员才能做到。实现一个复杂的编译系统（如 Java、C# 的编译器）的确是这样的。但编写一个完成一般任务的编译器或解释器就没那么复杂，甚至可以说是很容易。借助自动化工具（如本文介绍的 JavaCC），甚至可以在几个小时之内设计一种完整的编程语言。下文会结合 JavaCC 计算编写 Android 版本的计算器程序，该计算器可以计算表达式的值。

2 JavaCC 简介

JavaCC（Java Compiler Compiler）是目前非常流行的基于 Java 的分析器生成工具。JavaCC 本身是用 Java 语言实现的，而 JavaCC 生成的编译器源代码也是基于 Java 语言的。因此，JavaCC 正好可以用在 Android 上。

JavaCC 有自己的语法规则，只要按着语法规则（比编程语言简单得多，有些类似于正则表达式）编写要生成的语言分析器的规则，就可以利用 JavaCC 在一瞬间魔术般地生成一套分析器的源代码。

JavaCC 除了可以在命令行中完成工作外，还带了一个 Eclipse 插件，可以将分析器生成功能与 Eclipse 结合。这样在 Eclipse 中就可以完成所有的工作了。

3 JavaCC 下载和安装

JavaCC 是免费开源的，可以访问下面的地址下载 JavaCC 的最新版本：

http://javacc.java.net

对压缩包解压后，会看到 bin、doc 和 examples 3 个目录。其中 bin 目录包含了 JavaCC 的 jar 文件（javacc.jar）和其他一些相关文件，doc 目录是官方文档，examples 目录是 JavaCC 自带的例子。在 Android 中使用 JavaCC 并不需要引用 javacc.jar，而只需要将 JavaCC 生成的分析器源代码复制到 Android 工程中即可。

JavaCC 的 Eclipse 插件可以从下面的地址下载：

http://eclipse-javacc.sourceforge.net

这个 Eclipse 插件集成了 JavaCC，因此，如果在 Eclipse 中安装了该插件，就不需要下载 JavaCC 的压缩包了。

该插件主要可以完成如下两个工作：

（1）通过菜单方式生成分析器源代码。

（2）以高亮方式显示 JavaCC 的规则文件内容。

4 JavaCC 的语法

JavaCC 的语法可以有如下几部分（必须按顺序出现）：

JavaCC 选项
PARSER_BEGIN(标识符)
 Java 代码
PARSER_END(标识符)
TOKENS(词法元素)
描述语法和语义的Java 代码

如果不指定 JavaCC 选项，JavaCC 则使用选项的默认值生成分析器的 Java 代码。这些选项可以对 JavaCC 生成的 Java 代码进行调整。例如，下面的代码包含了两个 JavaCC 选项。其中 JAVA_UNICODE_ESCAPE 为 true 表示生成的分析器在读入

来卡网出品
LAICAR.COM
shop35833438.taobao.com

要分析的字符串之前会处理 Unicode 编码中的“\u”字符。DEBUG_PARSER 为 true 表示分析器会输出调试信息。更详细的 JavaCC 选项请读者参阅 JavaCC 的官方文档：

```
options
{
  JAVA_UNICODE_ESCAPE = true;
  DEBUG_PARSER = true;
}
```

JavaCC 语法的第 2 部分是夹在 PARSER_BEGIN 和 PARSER_END 之间的 Java 代码。其中 PARSER_BEGIN 和 PARSER_END 后面要跟同样的标识符号，标识符号就是要生成的分析器的主文件名。例如，如果是 PARSER_BEGIN (MyParser) 和 PARSER_END (MyParser)。JavaCC 会生成以下 3 个包含标识符 (MyParser) 的文件：

(1) MyParser.java: 分析器的主文件。

(2) MyParserTokenManager.java: Token 管理器，或称为词法分析器。

(3) MyParserConstants.java: 定义了在分析器中使用的常量。

除了这 3 个文件，还会生成一些其他的文件辅助 Java 源代码文件。

夹在 PARSER_BEGIN 和 PARSER_END 之间的 Java 代码主要包括分析器的主类 (包含在主文件中，如 MyParser.java) 以及 package，导入的包等。主类名应与 PARSER_BEGIN 及 PARSER_END 后面的标识符相同，而且 package 应与 *.jj 文件所在的包名相同。否则生成的 Java 代码会由于不符合 Java 规范而无法编译。

JavaCC 语法中真正重要的是第 3 部分和第 4 部分。其中第 3 部分定义了待分析的语言包含的所有 TOKEN。每一个 TOKEN 是一个语言的最基本的单元。例如，有一个条件语句如下：

```
if(x > 20)
    return;
else
   y = 30;
```

在分析条件语句时需要先将这语句拆成若干个独立的基本单元 (是不可再分的)。这个条件语句可被拆成如下的基本单元：

```
if ( x > 20 ) return ; else y = 30 ;
```

每一个基本单元就是一个 TOKEN。这些 TOKEN 会被一个个地送入语法分析器。也就是说，词法分析器读取的是一个个的字符，返回的是一个个 TOKEN。而语法分析器从词法分析器中读取 TOKEN，然后形成各种语法，并执行相应的语义动作。

JavaCC 语法的最后一部分描述了语法和语义。实际上这部分就是将语法产生式转换成了 Java 代码，并通过大括号将语义动作括了起来。下面会使用一个完整的例子介绍如何使用 JavaCC 计算表达式的值。

5 生成计算表达式的 Java 源代码

现在利用 JavaCC 编写一个完整的计算器程序。界面效果如图 1 所示。通过按屏幕上的数字、运算符等按钮，在屏幕最上方的显示区域输出这些数字和运算符。在输入完表达式后，按“=”按钮，会在表达式的下方显示运算结果。



图 1 计算器主界面

本例实现的计算器的核心是如何来计算表达式。这个计算器支持的表达式除了支持标准的四则运算外，还支持如下的运算符、常量和函数：

(1) %：取余。

(2) (：左括号。

(3))：右括号。

(4) .：小数点。

(5) PI：圆周率 (π)，值为 3.1415926。

(6) sin：正弦函数。

(7) cos：余弦函数。

编写 JavaCC 规则的第 1 步就是要确定分析器要分析的语言中的 TOKEN。表达式的大多数 TOKEN 都很简单，唯一复杂的就是数字。表达式不仅支持整数和浮点数，还支持用科学计数法表达很大或很小的数字。下面看一下数字的巴克斯范式 (BNF)：

```
// 数字的 BNF
number ::= [0-9]([0-9])* |
                  [0-9]+ '.' ([0-9])+ exponent? |
           '.' ([0-9])+ exponent? |
          ([0-9])+ exponent?
// 科学计数法的 BNF
exponent ::= [e,E]([+,-])?([0-9])+
```

BNF 类似正则表达式，是对拥有一定规则的字符串集合的抽象表达方式。在上面的 BNF 中使用了竖线 (|)，这个符号表示“或”，例如，number 使用了 3 个竖线，表示数字由如下 4 种情况组成。

(1) 整数：[0-9] ([0-9]) *。

(2) 小数点前有数字的浮点数 (支持科学计数法)：[0-9] + '.' ([0-9]) + exponent?。

(3) 小数点前没有数字的浮点数 (支持科学计数法)：'.'

([0-9]) + exponent?。

(4) 整数 (这些整数可能是科学计数法中 E 前面的数字): ∵ ([0-9]) + exponent?。

除了竖线外，BNF 还使用了加号 (+)、星号 (*)、中括号 ([…])、问号 (?)。这些符号的含义如下：

(1) 加号 (+)：表示元素至少重复 1 次。例如，a+表示 a 至少重复 1 次，也就是说 a、aa、aaa 都满足 a+规则。

(2) 星号 (*)：表示元素重复 0 次或多次。例如，a* 表示 ε| a+，其中 ε 表示空 (也可理解为空串)。a* 完全包含 a+，但还可以生成 ε。

(3) 中括号 ([…])：表示区间内元素可以任选一个。例如，[a,b] 表示 a | b。如果区间的字符有规律，可以用 "-" 连接首尾两个字符，如表示从 0 至 9 的数字中任选一个，可以用 [0 - 9]。

(4) 问号 (?)：表示元数重复 0 次可 1 次。例如，a?表示 ε| a。

在了解如何用 BNF 描述数字后，就可以在 JavaCC 规则文件中编写第 1 部分代码了。这部分代码主要定义表达式的 TOKEN：

```
TOKEN :
{
  < ADD : "+" >
| < SUB : "-" >
| < MUL : "*" >
| < DIV : "/" >
| < MOD : "%" >
| < LPAREN : "(" >
| < RPAREN : ")" >
| < NUMBER :
    [ "0"-"9" ] ([ "0"-"9" ])*
  | ([ "0"-"9" ])+ "." ([ "0"-"9" ])* (< EXPONENT >)?
  | "." ([ "0"-"9" ])+ (< EXPONENT >)?
  | ([ "0"-"9" ])+ < EXPONENT > >
| < #EXPONENT : [ "e", "E" ] ([ "+", "-" ])? ([ "0"-"9" ])+ >
| < SIN : "sin" >
| < COS : "cos" >
| < PI : "PI" >
}
```

所有的 TOKEN 都要放在 TOKEN : {…} 中。如果有多个 TOKEN，中间用竖线分隔。每一个 TOKEN 都要放在尖括号 (<…>) 中。TOKEN 分成两部分：分析器使用的常量和要扫描的字符，这两部分中间用冒号 (:) 分隔。例如，加号的 TOKEN 是<ADD : " +" >，其中 ADD 是分析器使用的常量，而 "+" 是要扫描的字符。如果分析器扫描到 "+" 后，会用 ADD 表示这个符号。

在书写表达式时为了更方便人们阅读，往往会在不同 TOKEN 之间加上若干个空格、制表符、回车符或换行符。这些字符可以起到如下的作用：

(1) 使表达式看起来更优美。

(2) 分隔不同的 TOKEN。被这些符号分开的字符串会被分析器单独分析，分析的结果不会连接起来。例如，"si n" 中间用空格分隔，尽管分析器会将空格过滤掉，但并不会将 si 和 n 连接起来形成 sia。

由于这些符号并不参与计算，因此，在分析表达式时需要将这些字符过滤掉。所有要过滤掉的字符要放在 SKIP : {…} 中，代码如下：

```
SKIP :
{
  " "
| "\t"
| "\r"
| "\n"
}
```

下面来编写 JavaCC 规则中最重要的一部分：语法和语义动作描述。这一部分是标准的 Java 代码。

在编写代码之前，先看一下表达式的 BNF，每一行也可称为产生式：

```
expr ::= term ((+|-) term)*
term ::= factor ((*|/|%) factor)*
factor ::= number | (expr) | sin(expr)  | cos(expr) | PI
```

运算符存在优先级，因此，在写 BNF 时应注意这一点。由于 *、/、%的优先级相同，但高于+和-。因此，应将包含 *、/、%的产生式作为一个整体 (term) 在包含+和-的产生式中使用。由于上面的 BNF 由 3 个产生式组成。这些产生式由两部分组成：终结符和非终结符。终结符就是具体待扫描的字符，如+、-、*、PI 等。而非终结符就是英文单词，如 expr、term、factor 等。将 BNF 转换成 Java 代码的原则是每一个非终结符对应一个方法。这 3 个产生式共有 3 个非结符符，因此，需要使用 3 个方法 (expr、term 和 factor) 来转换上面的 BNF 表达式。下面是转换后的 Java 代码：

```
// expr ::= term ((+|-) term)* 对应的 expr 方法
double expr() :
{
  // 这里面定义的是临时变量,这些变量一般用于保存语义部分产生的值
}
{
  first = term(){ // 花括号中的是语义部分 }
  (
     < ADD > term(){ … }|
       < SUB > term(){ … }
  )*{ … }     // ((+|-) term)*
}
// term ::= factor ((*|/|%) factor)*  对应的 term 方法
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
double term() : { }
{
  first = factor(){ … }
  (
    < MUL > factor(){ … }|
      < DIV > factor(){ … }|
      < MOD > factor(){ … }
   )*{ … }  // ((*|/|%) factor)*
}
  // factor ::= number | (expr) | sin(expr)  | cos(expr) | PI
  // 对应的 factor 方法
  double factor() : { }
  {
    < NUMBER >{ … } |   // number
     < LPAREN > expr() < RPAREN >{ … }|  // (expr)
  < SIN > < LPAREN > expr() < RPAREN >{ … }| // sin(expr)
< COS > < LPAREN > expr() < RPAREN >{ … }| // cos(expr)
     < PI >{ … }       // PI
  }
```

从上面的代码可以看出。转换的过程只是机械地进行了翻译。如果遇到非终结符，直接将其替换成对应的方法（如 term ()），如果遇到终结符，就将其转换成 TOKEN 对应的常量，如+替换成< ADD >、sin 替换成< SIN >等。JavaCC 根据上面的代码生成的分析器的确可以分析表达式，但除此之外，什么都做不了。也就是说，分析器根本不可能用于计算表达式的值。要想生成一个可以计算表达式的值的分析器，就要在上面的代码中添加语义部分。

语义部分是标准的 Java 代码，需要放在花括号中。上面代码中的 3 个方法有如下两个地方需要写语义代码。

（1）冒号后面的花括号中：在这里一般要定义语义代码中要使用的变量。例如 double factor () : {double temp = 0; }。

（2）语法描述后面的大括号中：这里是真正的语义动作代码。需要根据实际情况编写不同的 Java 代码。例如，< LPAREN > temp = expr () < RPAREN > { return temp; }

除此之外，在语法规则部分还需要将中间结果保存在变量中，例如，temp=expr ()。下面看一下完整的语法和语义的代码：

```
  // expr ::= term ((+|–) term)* 对应的 expr 方法
  double expr() :
  {
  double temp = 0;
  double first, second;
}
{
  first = term(){temp = first;}
  (
    < ADD > second = term()
    {
      temp = first + second;
      first = temp;
    }|
      < SUB > second = term()
    {
      temp = first – second;
      first = temp;
    }
  )*  // ((+|–) term)*
  {
    return temp;   // 返回最后的结果
  }
}
  // term ::= factor ((*|/|%) factor)*  对应的 term 方法
  double term() :
  {
    double temp = 0;
    double first, second;
  }
  {
    first = factor(){ temp = first; }
    (
      < MUL > second = factor()
      {
        temp = first * second;
// 这条语句必须得加,否则就只计算第 1 个和最后一个操作数了
        first = temp;
      }|
      < DIV > second = factor()
      {
        temp = first / second;
        first = temp;
      }|
        < MOD > second = factor()
      {
        temp = first % second;
        first = temp;
      }
    )*  // ((*|/|%) factor)*
    {
      return temp; // 返回最后的结果
    }
  }
  // factor ::= number | (expr) | sin(expr)  | cos(expr) | PI
  // 对应的 factor 方法
  double factor() :
  {
     double temp = 0;
  }
  {
    < NUMBER >
    {
      return Double.parseDouble(token.image);
    }
  | < LPAREN > temp = expr() < RPAREN >
```


LAICAR.COM
shop35833438.taobao.com

```
{
  return temp;
}
| < SIN > < LPAREN > temp = expr() < RPAREN >
{
  return java.lang.Math.sin(temp);
}
| < COS > < LPAREN > temp = expr() < RPAREN >
{
  return java.lang.Math.cos(temp);
}
| < PI >
{
  return 3.1415926;
}
}
```

现在来编写 JavaCC 规则代码的最后一部分。这部分用于定义了分析器主类、package、导入包以及 JavaCC 选项：

```
options
{
  static = false; // 不生成静态的变量,以便可以创建多个
//分析器实例
}
PARSER_BEGIN(CalcParser)
package mobile.android.jx.calc.parser;
import java.io.StringReader;
import java.io.Reader;
public class CalcParser
{
 // 允许使用分析器时直接指定 String 类型的表达式
 public CalcParser(String expr)
 {
   this ((Reader) (new StringReader(expr)));
 }
}
PARSER_END(CalcParser)
```

在 CalcParser.jj 文件上单击右键菜单，单击 “Compile with JavaCC” 菜单项，会生成若干个分析器源代码文件，例如，CalcParser.java、CalcParserConstants.java 等。

6 编写计算器的主程序

计算器主界面如图 1 所示。界面分成两部分：上半部的显示区域和下半部的按钮区域。按钮的排列使用 GridView 控件。其他的是 TextView 控件。布局代码如下：

```
main.xml
<?xml version="1.0" encoding="utf-8"?>
 <LinearLayout  xmlns:android = "http://schemas.android.
com/apk/res/android"
      android:orientati on = "vertical"  android:layout_width = "
fill_parent"
      android:layout_height="fill_parent">
      <! -- 显示表达式的区域  -->
       <TextView  android:id = "@ +id/textview_expression"
android:layout_width="fill_parent"
  android:layout_height="wrap_content" android:lines="3"
       android:textSize = "15sp"  android:textColor = "#FF0"
android:background="#00F" android:padding="5dp" />
      <! -- 显示计算结果的区域  -->
      <TextView android:id="@+id/textview_result" android:
layout_width="fill_parent"
   android:layout_height="wrap_content" android:lines="1"
        android:textSize = "20sp"  android:textColor = "#FF0"
android:background="#00F"
android:padding="5dp" android:layout_marginTop="6dp" />
<! -- 显示按钮的区域 -->
      <GridView android:gravity="center" android:id="@+id/
gridview_buttons"
   android:padding="10dp" android:layout_width="fill_parent"
             android:layout_height = "fill_parent"  android:
horizontalSpacing="10px"
    android:verticalSpacing = "10px"  android:stretchMode = "
columnWidth"
  android:columnWidth="80dp" android:numColumns="4" />
   </LinearLayout>
```

在 GridView 中显示的按钮信息由 GridAdapter 对象。GridAdapter 类的代码如下：

```
package mobile.android.jx.calc;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.Button;
public class GridAdapter extends BaseAdapter
{
   private LayoutInflater layoutInflater;
   private Context context;
   // 定义按钮的文本
   private final String[] buttonTexts = new String[]
{ "(", ")", "退格", "清除", "7", "8", "9", "+", "4", "5", "6", "-", "1", "2",
        "3", "*", "0", ".", "PI", "/", "sin", "cos", "%", "=" };
   public GridAdapter(Context context)
   {
      layoutInflater = (LayoutInflater) context
.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
      this.context = context;
   }
   @Override
   public int getCount()
   {
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
    return buttonTexts.length;
}
@Override
public Object getItem(int position)
{
    return null;
}
@Override
public long getItemId(int position)
{
    return 0;
}
// 获取每一个按钮的 Button 对象
@Override
public View getView (int position, View convertView,
ViewGroup parent)
{
    Button button =  (Button) layoutInflater.inflate(R.
layout.button, null);
    // 设置按钮文本
    button.setText(buttonTexts[position]);
    // 设置按钮的单击事件
  button.setOnClickListener((OnClickListener) context);
    return button;
}
}
```

在 GridAdapter.getView 方法中设置了每一个按钮的单击事件，而单击事件所在的类就是创建 GridAdapter 对象时传入的 context。因此，创建 GridAdapter 对象的 Activity 要实现 OnClickListener 接口。按钮的单击事件代码如下：

```
public void onClick(View view)
{
    Button button = (Button) view;
    // 获取按钮的文本
    String text = button.getText().toString();
    // 单击"="按钮
    if (text.equals("="))
    {
        // 根据表达式创建 CalcParser 对象(CalcParser 是
//分析器的主类)
CalcParser parser = new CalcParser(tvExpression.getText()
            .toString());
    try
    {
        // 计算结果,并将结果显示在 TextView 控件中
  tvResult.setText("= " + String.valueOf(parser.expr()));
    }
    catch (Exception e)
    {
        // 输出错误信息
        tvResult.setText(e.getMessage());
        }
    }
    // 按"退格"按钮
    else if (text.equals("退格"))
    {
        SimpleCharStream simpleCharStream = new
SimpleCharStream(
    new StringReader(tvExpression.getText().toString()));
        // 直接使用 CalcParserTokenManager 对象获取组
//成表达式的
        // 所有 TOKEN
      CalcParserTokenManager calcParserTokenManager
= new CalcParserTokenManager(simpleCharStream);
        String s = "";
        String first = "";
        String second = "";
    first = calcParserTokenManager.getNextToken().image;
        // 每次退格只删除最后一个 TOKEN
        while (! first.equals(""))
        {
            second = first;
  first = calcParserTokenManager.getNextToken().image;
            if (! first.equals(""))
                s += second;
        }
        tvExpression.setText(s);
    }
    // 单击"清除"按钮
    else if (text.equals("清除"))
    {
        tvExpression.setText("");
        tvResult.setText("");
    }
    // 单击其他按钮
    else
    {
        tvExpression.append(text);
    }
}
```

7 结语

着重讨论了编译理论在 Android 的应用。编译原理虽然学起来很枯燥，但通过很多辅助软件的帮助，几乎不需要太高深的理论知识就可以在很短的时间内设计出一种完整的编程语言。编译器自动生成工具有很多。主要介绍了可以生成 Java 代码的 JavaCC。因为这个工具可以很好地与 Android 结合。最后给出了一个完整的 JavaCC 与 Android 结合使用的计算器程序的例子，以便于在 Android 中使用JavaCC。

（收稿日期：2013-01-10）

shop35833438.taobao.com

模拟退火算法求解二次规划问题与实现

王文举

摘　要：介绍了模拟退火算法的基本原理、设计思路和在求解二次规划问题中的具体应用，并给出了完整的代码，对学习和应用模拟退火算法有很好的借鉴作用。

关键词：C# 语言；模拟退火算法；二次规划问题

1　引言

模拟退火算法来源于固体退火原理，它是由 Kirkpatrick 最早提出，基于 Monte-Carlo 迭代求解策略的一种随机寻优算法，其出发点是基于物理中固体物质的退火过程与一般组合优化问题之间的相似性。模拟退火算法从某一较高初温出发，伴随温度参数的不断下降，结合概率突跳特性在解空间中随机寻找目标函数的全局最优解，即在局部最优解能概率性地跳出并最终趋于全局最优。

二次规划是非线性规划中比较简单的一类，很多方面的实际问题都可以抽象成二次规划的模型去求解，例如在运筹学中，它被广泛用于经济调度、合理分配、计划决策等问题。传统解决二次规划问题的方法过程比较复杂，计算时间较长，使其在大范围优化中的应用受到限制。文中采用模拟退火算法求解二次规划问题，具有计算速度快和精度高的特点。



图 1　模拟退火算法求解二次规划问题

2　设计思路

如图 1 所示，考虑如下二次规划问题，求解下列函数的最小值，xx+yy+xy−4x−6y，约束条件为 x+y=5，并且 x,y 都大于零。模拟退火算法的实现过程如下：

（1）初始化：初始温度 Temperature，初始解状态 PreX，对于每个温度迭代次数 MarkovLength。

（2）对于 i = 0…MarkovLength，求解第 (3) 至第 (6) 步。

（3）产生新解 NextX。

（4）计算增量△= GetFitness (PreX) − GetFitness (NextX)，其中 GetFitness (double x) 为评价函数。

（5）如果△大于零，则接受 NextX 作为新的当前解，否则以概率 exp (−△/ Temperature) 接受 NextX 为新的当前解。

（6）温度逐渐减小，转到第（2）步。

（7）如果满足终止条件则输出最优解，结束程序。

网络资料上针对模拟退火算法有很详细的介绍，但是缺乏翔实的代码实现，对于初学者很难理解算法和实现，下文给出基于 C# 语言的具体实现代码，读者可以借鉴使用。

3　实现代码

3.1　窗口类代码

```
public partial class Form1 : Form
   {  SimulateAnnealing mySimulateAnnealing; //模拟退火算
//法类
      public Form1()
      {  InitializeComponent();
         mySimulateAnnealing = new SimulateAnnealing();
         mySimulateAnnealing.richTextBox1 = richTextBox1;
         InitialRichBox();
      }
      private void btnRun_Click(object sender, EventArgs e)
      {  InitialRichBox();
      if (mySimulateAnnealing.SA() == true) //模拟退火算法
         {  MessageBox.Show("取得最优解");
            double best = mySimulateAnnealing.BestX;//x 值
             double f = mySimulateAnnealing.GetFitness
(best);//函数最小值
            richTextBox4.Text = "x 取值为:" + best.ToString()
+ "\r\ny 取值为:" +
                                Convert.ToString(5 – best) +
      "\r\n 函数最小值为:" + f.ToString() + "\r\n";
      }
      else
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
        {  MessageBox.Show("没有找到最优解,请重新运行
程序");
        }
    }
    private void InitialRichBox()
    {  richTextBox1.Text = "";
       richTextBox2.Text = "x*x+y*y+xy-4x-6y";//二次规        //点开始
//划的问题
   richTextBox3.Text = "x+y=5 并且 x>0,y>0";//约束条件
       richTextBox4.Text = "";
    }
```

3.2 模拟退火算法类代码

```
class SimulateAnnealing
  {  const double XMIN = 0; //搜索区间最大值
     const double XMAX = 5; //搜索区间最小值
     //冷却表参数
     int MarkovLength = 1000;//马可夫链长度
     double DecayScale = 0.95;//衰减参数
     double StepFactor = 0.02;//步长因子
     double Temperature = 100;//初始温度
     double Tolerance = 1e-8;//容差
     double PreX, NextX; //上一个、下一个 X 值
     double PreBestX;//上一个最优解
     public double BestX;//最终解
   double AcceptPoints = 0.0;//Metropolist 过程中总接受点
     Random rnd = new Random();
     public RichTextBox richTextBox1;
      public double GetFitness(double x) //根据二次规划问
//题,求解适应度
     {  double fit = 1.0;
        fit = x * x + (5 - x) * (5 - x) + x * (5 - x) - 4 * x -
6 * (5 - x);
        return fit;
     }
     public bool SA()//模拟退火算法
     {  //随机选点
        PreX = XMAX * rnd.NextDouble();
        PreBestX = BestX = PreX;
        //每迭代一次退火一次(降温),直到满足迭代条件为止
        do
        {  Temperature *= DecayScale;
        AcceptPoints = 0.0;
        //在当前温度 T 下迭代马可夫链长度次数
        for (int i = 0; i < MarkovLength; i++)
        {  //在此点附近随机选择下一点
           do
           {  NextX = PreX + StepFactor * XMAX * (rnd.
NextDouble() - 0.5);
           } while (! (NextX > XMIN && NextX < XMAX));
           //是否全局最优解
           if (GetFitness(BestX) > GetFitness(NextX))
           {  PreBestX = BestX; //保留上一个最优解
              BestX = NextX; //此为新的最优解
           }
           //Metropolist 过程
           if (GetFitness(PreX) - GetFitness(NextX) > 0)
           {  //目标代价变化,下一个迭代的点以新接受的

              PreX = NextX;
              AcceptPoints++;
           }
           else
           {  //当前温度下状态活动概率,模拟退火的核心
//公式
              double change = -1 * (GetFitness(NextX) -
GetFitness(PreX)) / Temperature;
              if (Math.Exp(change) > rnd.NextDouble())
              {  PreX = NextX;
                 AcceptPoints++;
              }
           }
        }
        richTextBox1.Text += " 温度为:" + Temperature.
ToString() + "\r\nx 取值为:" + PreX.ToString() + "\r\ny 取值为:"
+ Convert.ToString(5 - PreX) +
                    "\r\n 函数最小值为:" + GetFitness
(PreX).ToString() + "\r\n";
     } while (Math.Abs(GetFitness(BestX) - GetFitness
(PreBestX)) > Tolerance);
     richTextBox1.Text += " 温度为:" + Temperature.
ToString() + "\r\nx 取值为:" + BestX.ToString() + "\r\ny 取值为:"
+ Convert.ToString(5 - BestX) +
                    "\r\n 函数最小值为:" + GetFitness
(BestX).ToString() + "\r\n";
     return true;
  }
}
```

4 结语

模拟退火算法是一种通用的优化算法，通过赋予搜索过程一种时变且最终趋于零的概率突跳性，可有效避免陷入局部最优并最终趋于全局最优，在理论上算法具有概率的全局优化性能，目前已经在生产调度、控制工程、机器学习、神经网络、信号处理等领域得到了广泛应用。

（收稿日期：2013-02-12）

shop35833438.taobao.com

利用 VB 开发 Flash Cookies 清除器

华 云

摘 要：利用 API 函数及 FSO 文件系统对象，开发了 Flash Cookies 清除器。

关键词：Visual Basic 语言；Windows API 函数；FSO 文件系统对象；Flash Cookies 处理

1 引言

最近，随着央视 315 晚会的播出，Flash Cookies 这个词在互联网上被抄得沸沸扬扬的，迅速的成为了热门话题。在 315 晚会上，某销售人员得意地说 Flash Cookie 比普通 Cookie 更为靠谱，因为用户无法删除它们。其实这个说法并不完全准确，用户只需要一些简单的操作，就可以清除这些信息。这些操作对于一般非熟练用户来说，并不十分容易。基于此，本文介绍利用 VB 开发一个 Flash Cookies 清除器，从而实现“一点即清”。

2 原理

Flash Cookie，指的是“Flash Player 本地存储”。即用户访问特定网站时可以在用户的计算机上创建该存储信息的文件。它们可以像浏览器 Cookie 那样用来存储信息。例如登录数据或用户访问的网页等。

Flash Cookies 一般保存在当前用户的“Application Data\Macromedia\Flash Player”文件夹下，扩展名为 sol 的文件。通过程序首先获取该文件夹，再遍历获取这些 sol 文件，即可实现清除 Flash Cookie。

3 界面设计



图 1 程序运行效果

进入 VB，选择“工程”菜单下的“部件”，在弹出的菜单中选择“Microsoft Windows Common Controls 6.0”。然后，在“工程”菜单中选择“引用”，在弹出的对话框中选择“Microsoft Scripting Runtime”，将引用 FSO 对象模型对文件进行操作。在默认窗体 Form1 上添加如下控件：两个文本框控件 Text1、Text2；一个 Listview1 控件，设置其 Checkboxes 属性为 True；一个状态栏控件 Statusbar1，并为它插入两个窗格；再加入 6 个命令按钮 Command1-Command6，其 Caption 属性分别设置为“搜索 cookies”、“全选”、“反选”、“取消选择”、“删除所选文件”、“复制所选文件”；设置完成后运行的效果如图 1 所示。

4 源代码

现在切换到代码编辑器输入以下代码，其中的 API 函数及常数声明可以从 API 浏览器中复制：

```
Private Const MAX_PATH = 260
Public index As Integer
Private Declare Function SHGetSpecialFolderLocation Lib "
shell32" (ByVal hWnd As Long, ByVal nFolder As Long, Pidl
As Long) As Long
Private Declare Function SHGetPathFromIDList Lib "shell32"
(Pidl As Long, ByVal FolderPath As String) As Long
Public Enum SYSTEM_FOLDERS
   CSIDL_APPDATA = &H1A 'Current User\Application Data
 End Enum
Private Declare Function SHBrowseForFolder Lib "shell32.dll"
Alias "SHBrowseForFolderA" (lpBrowseInfo As BROWSEINFO)
As Long
Private Type BROWSEINFO
    hOwner As Long
    pidlRoot As Long
    pszDisplayName As String
    lpszTitle As String
    ulFlags As Long
    lpfn As Long
    lParam As Long
    iImage As Long
End Type
'自定义查找文件函数
Sub FindFile(fd As Folder, FileName As String)
   Dim sfd As Folder, f As File
   'Part I?查找该文件夹的所有文件
   For Each f In fd.Files
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
StatusBar1.Panels(2).Text = f.Path
StatusBar1.Panels(1).Text = "请稍候,正在搜索..."
 If UCase(f.Name) Like UCase(FileName) Then
  ' 名称
  ListView1.ListItems.Add index, , f.Name
  ' 大小
  ListView1.ListItems(index).SubItems(1) = f.Size
  ' 路径
  ListView1.ListItems(index).SubItems(2) = fd.Path
    ' 类型
    ListView1.ListItems(index).SubItems(3) = f.Type
    ' 修改日期
            ListView1.ListItems (index).SubItems (4) = f.
DateLastModified
    ' 创建日期
    ListView1.ListItems(index).SubItems(5) = f.DateCreated
    index = index + 1
   End If
   DoEvents
  Next
  ' Part II?循环查找所有子文件夹
  For Each sfd In fd.SubFolders
   FindFile sfd, FileName ' 循环查找
  Next
End Sub
' 获取系统特殊文件夹位置函数
Private Function FindSystemFolder (ByVal lngNum As Long)
As String
  Dim lpStartupPath As String * MAX_PATH
  Dim Pidl As Long
  Dim hResult As Long
  hResult = SHGetSpecialFolderLocation(0, lngNum, Pidl)
  ' 如果出错, 返回 0
  If hResult = 0 Then
   ' 如果没有错误,提取路径保存到 lpStartupPath 变量中
   hResult = SHGetPathFromIDList(ByVal Pidl, lpStartupPath)
   If hResult = 1 Then
     lpStartupPath = Left $ (Trim $ (lpStartupPath), InStr
(lpStartupPath, Chr(0)) - 1)
    FindSystemFolder = Trim$(lpStartupPath)
   End If
  End If
End Function
Private Sub Command1_Click()
' 搜索
 ListView1.ListItems.Clear
 index = 1
 Dim fs As New FileSystemObject ' 建立 FileSystemObject
 Dim fd As Folder  ' 定义 Folder 对象
 Dim sfd As Folder
  ' 获取当前用户的 application data 文件夹位置
 fdir = FindSystemFolder(CSIDL_APPDATA)
  ' 设置 flash cookies 搜索路径
 fdir = fdir & "\Macromedia\Flash Player"
 Set fd = fs.GetFolder(fdir)
 Screen.MousePointer = vbHourglass
 FindFile fd, "*.sol"
 Screen.MousePointer = vbDefault
 ' 设置状态栏文本
 StatusBar1.Panels (2).Text = " 共有" & ListView1.ListItems.
Count & "个文件"
 StatusBar1.Panels(1).Text = "就绪"
End Sub
' 反选
Private Sub Command2_Click()
 For i = 1 To ListView1.ListItems.Count
   ListView1.ListItems(i).Checked = Not ListView1.ListItems
(i).Checked
  Next
End Sub
' 删除所选文件
Private Sub Command3_Click()
 Dim fs As New FileSystemObject
 For i = 1 To ListView1.ListItems.Count
  If ListView1.ListItems(i).Checked Then
    fs.DeleteFile ListView1.ListItems (i).SubItems (2) & "\" &
ListView1.ListItems(i).Text
  End If
 Next
 Dim ListCount As Long
 ListCount& = ListView1.ListItems.Count + 1
 Do While ListCount& > 1&
  ListCount& = ListCount& - 1
  If ListView1.ListItems(ListCount&).Checked = True Then
   ListView1.ListItems.Remove ListCount&
  End If
 Loop
End Sub
Private Sub Command4_Click()
' 全选
 For i = 1 To ListView1.ListItems.Count
  ListView1.ListItems(i).Checked = True
 Next
End Sub
Private Sub Command6_Click()
  Dim bi As BROWSEINFO ' 声明必要的变量
  Dim rtn&, Pidl&, Path$, pos%
  bi.hOwner = Me.hWnd ' 使对话框处于屏幕中心
  bi.lpszTitle = "选择目标文件夹..." ' 设置标题文字
  bi.ulFlags = BIF_RETURNONLYFSDIRS ' 返回文件夹的类型
  Pidl& = SHBrowseForFolder(bi) ' 显示对话框
  Path = Space(512) ' 设置字符数的最大值
  T = SHGetPathFromIDList(ByVal Pidl&, ByVal Path) ' 获得所
' 选的路径
```


LAICAR.COM
shop35833438.taobao.com

```
pos% = InStr(Path$, Chr$(0)) ' 从字符串中提取路径
SpecIn = Left(Path$, pos - 1)
If Right$(SpecIn, 1) = "\" Then
  specout = SpecIn
Else
  specout = SpecIn + "\"
End If
Dim fs As New FileSystemObject
For i = 1 To ListView1.ListItems.Count
  If ListView1.ListItems(i).Checked Then
    fs.CopyFile ListView1.ListItems (i).SubItems (2) & "\" &
ListView1.ListItems(i).Text, specout, True
  End If
 Next
End Sub
Private Sub Form_Load()
 ListView1.View = lvwReport ' 设置列表框查看模式
 ' 设置列表框的标题
 ListView1.ColumnHeaders.Add , , "文件名称"
 ListView1.ColumnHeaders.Add , , "大小"
 ListView1.ColumnHeaders.Add , , "所在文件夹 "
 ListView1.ColumnHeaders.Add , , "类型"
 ListView1.ColumnHeaders.Add , , "修改日期"
 ' 设置初始值
 index = 1
 StatusBar1.Panels(1).Text = "就绪"
End Sub
' 单击列表项显示文件位置及内容
Private Sub ListView1_ItemClick (ByVal item As
MSComctlLib.ListItem)
   On Error Resume Next
   Dim FileNum As Integer
   FileNum = FreeFile
    Open ListView1.SelectedItem.SubItems (2) & "\" &
ListView1.SelectedItem.Text For Input As #FileNum
   Line Input #FileNum, txtstr
   Close #FileNum
   Text3.Text = txtstr ' 文件内容
Text1.Text = ListView1.SelectedItem.SubItems(2) ' 文件位置
End Sub
```

5 结语

程序在 VB6.0 中文企业版，Windows XP 环境下运行通过，程序稍加修改即可实现对其他类型文件的搜索及删除。

（收稿日期：2013-03-20）

（上接第 15 页）

从以上这些例子可以看出，使用 Web3.0 的语义网技术确实能够对查询效果的提高起到很大的作用，可以预见在未来，该技术将大大地推进互联网的发展。

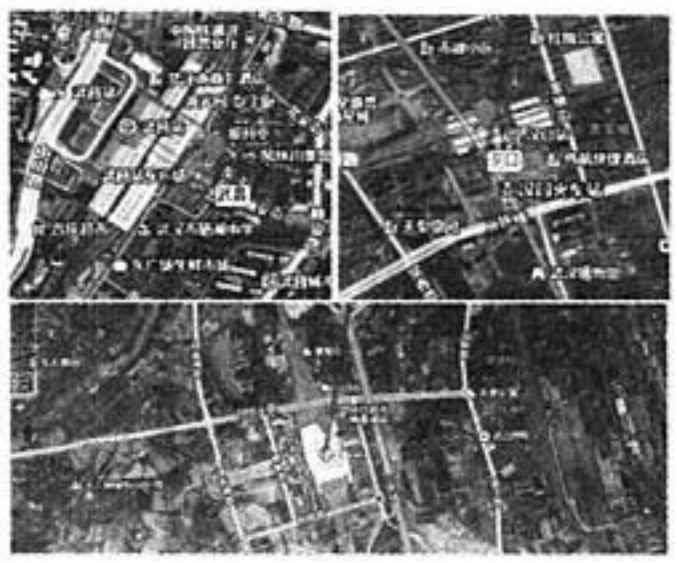


图 5 模糊查询到的武汉地区三大火车站



图 6 精确输入时，查询结果自动适应到查询结果

5 结语

介绍了基于 C# 环境中的 Jena 实现关系数据库中创建本体与规则推理。利用 Jena，笔者完成了基于关系数据库的本体自动创建、本体逻辑推理及查询的功能。从本篇实现的功能可以看出，Jena 的执行效率较高、能够方便地操作本体，但是在某些方面还是有缺点的，比如内存读入 rules 的 bug；而且当创建本体的数据量大时，程序运行会比较缓慢等等。Jena 是开源代码，因此程序员可以直接在源程序上进行修改，随着时光的流转，Jena 将会越来越成熟。

参考文献

[1] 董志. C# 集成 Google Map API 进行地理空间的定位 [J]. 电脑编程技巧与维护，2011 (19)：47-53.

[2] Michael Bauman. Web3.0: The next step for the Internet [J]. Information Today，2009，26 (5)：46.

[3] Sean Bechhofer, Frank van Harmelen and Jim Hendler et al. OWL Web Ontology Language Reference. W3C Recommendation 10 February 2004.

[4] Jennifer Golbeck, Gilberto Fragoso, Frank Hartel and Jim Hendler et al. The national cancer institute' s thesaurus and ontology. Journal of Web Semantics, 1 (1) , 2003.

[5] http://www.ibm.com/developerworks/cn/java/j-jena/.

（收稿日期：2013-01-24）

来卡网出品
LAICAR.COM
shop35833438.taobao.com

VBA 编程实现燃气－蒸汽联合循环性能计算

卞 晶　马 贤　马 勇

摘　要： 结合实例，介绍了 Excel VBA 的用法。用 VBA 编写了计算燃气–蒸汽联合循环性能计算的代码及窗体文件等。使 VBA 的应用领域拓展到我国的发电领域。

关键词： Excel 软件；VBA 编程；联合循环；发电；燃气；蒸汽

1　引言

近年来 Microsoft Office 中 Excel 的功能越来越强大。Excel 中的 VBA（Visual Basic for Application）是 Microsoft 公司长期追求的目标，使可编程应用软件得到完美的实现，它作为一种通用的宏语言可被所有的 Microsoft 可编程应用软件所共享。在工程技术领域特别是发电厂运行中，电厂的运行人员需要知道当前运行的燃气–蒸汽联合循环的实时性能参数，可以借助于 Excel VBA 的强大功能来编程实现。

为了对所编程序的对象有一个初步的了解，可以看燃气–蒸汽联合循环机组的运行系统简图，如图 1 所示。

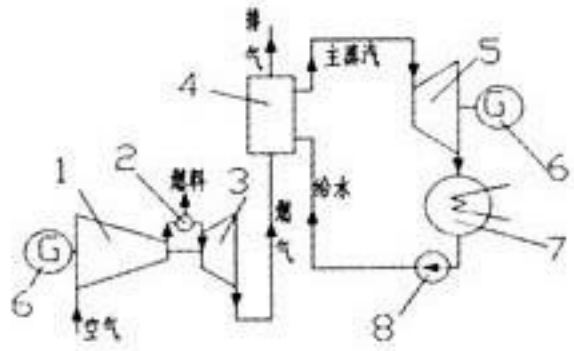


图 1　燃气–蒸汽联合循环发电方案

注：1—压气机；2—燃烧室；3—燃气透平；4—余热锅炉；5—蒸汽透平；6—发电机；7—凝汽器；8—给水泵。

空气进入压气机里面经过压缩机之后成为高压空气，在燃烧室中吸收大量热量，然后在燃气透平中膨胀做功。做完功的燃气进入余热锅炉中放热加热水蒸汽，水蒸汽吸收热量之后在蒸汽透平中膨胀做功带动发电机发电。在蒸汽透平中膨胀做功后的蒸汽在冷凝器中冷凝成水然后继续在余热锅炉中吸热。这里采用美国 ASME PTC47 整体煤气化联合循环电站性能试验规程的规定中提供的算法进行编程，在该算法中不考虑修正的影响。

2　程序设计

2.1　编程环境

新建一个 Microsoft Excel 2010 文档。打开该文档，另存为“Excel 启用宏的工作簿”，并将文档名称修改为“燃气–蒸汽联合循环性能计算”，如图 2 所示。

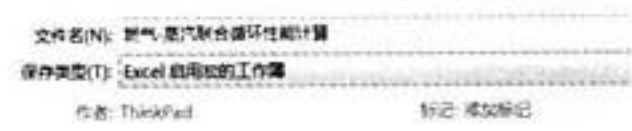


图 2　启用宏的工作薄

在打开的 Excel 文档中，按“Alt+F11”即进入 VBA 编程环境。

2.2　代码及窗体

在打开的编程环境中，插入窗体，如图 3 所示。

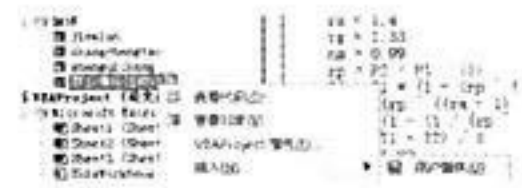


图 3　插入窗体

在新建的窗体上编辑各控件，编辑好后的窗体如图 4 所示。



图 4　窗体

图 4 中各个文字框的名称可以直接设置为对应的变量名。如燃料温度的文字框可以在文字框的属性窗口中把名称直接设置成变量名 Tf。设置的方式可以如图 5 所示。

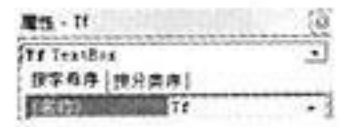


图 5　属性窗口设置变量名

设置好这些之后，就可以在“开始计算”的按钮下输入程序代码了。输入程序代码的方式是在“开始计算”的按钮上双击鼠标左键。双击鼠标左键之后，这个按钮的事件函数弹出。“开始计算”按钮的事件函数如下：

```
Private Sub 开始计算 _Click()
End Sub
```

在这个函数中，输入代码之后，完整的事件函数如下：

```
Private Sub 开始计算 _Click()
```


LAICAR.COM
shop35833438.taobao.com

```
'1 压气机计算
Dim ra, rg, rp, Rpa, Rpg, nm, Cpa, Ta As Double
    ra = 1.4
    rg = 1.33
    nm = 0.99
    rp = P2 / P1  '(1)
    T2 = T1 * (1 + (rp ^ ((ra - 1) / ra) - 1) / nc) '(3)
    Rpa = (rp ^ ((ra - 1) / ra) - 1) / nc
    Rpg = (1 - (1 / (rp ^ ((rg - 1) / rg))))
    Ta = (T1 + T2) / 2
    Cpa = 1.02
    Wc = Cpa * T1 * Rpa / nm
'2 燃烧室计算
Dim mf, Cpg As Double
    Cpg = 1.8083 - 0.0023127 * TIT + 0.000004045 * TIT *
TIT - 0.0000000017363 * TIT * TIT * TIT
    mf = ma * (Cpg * TIT - Cpa * T1 * (1 + Rpg)) / (LHV +
Cpf * Tf - Cpg * TIT)
    f = mf / ma
    sita = TIT / T1
'3 燃气透平计算
Dim T3, T4, Wt, WGnet, P, SFC, Qadd, nth, HR As Double
    T3 = TIT
    T4 = T3 * (1 - nt * Rpg)
    Wt = Cpg * TIT * nt * Rpg / nm
    WGnet = Wt - Wc
    P = ma * WGnet
    Debug.Print "P="; P
    SFC = 3600 * f / WGnet
    Qadd = f * LHV
    nth = WGnet / Qadd
    HR = 3600 / nth
Debug.Print "HR="; HR
'4 蒸汽轮机计算
Dim Qav, ms, Wst As Double
Dim Tg1, Tg4 As Double
Dim h1f As Double
Dim mg As Double
    Tg1 = T4
    Tg4 = 393.15
    h1f = 0.985
    mg = ma + mf
    Qav = mg * Cpg * (Tg1 - Tg4) * h1f
    ms = Qav / (hsh - hw1)
    Wst = ms * (hsh - hn)
'5 总参数
    W = P + Wst
    Debug.Print "W"; W
    nall = (P + Wst) / (LHV * mf)
    Debug.Print "nall="; nall
    HRt = 3600 / nall
    Debug.Print "HRt="
End Sub
```

该段程序代码中，如果用来注释的文字可以在文字前加“'”，注意这个单引号是英文的单引号，不是中文中的单引号，例如“'5 总参数”所示。另外程序中的“Debug. Print “nall=”; nall”是在立即窗口中显示变量 nall 的值。在立即窗口中显示的变量并不影响窗体的正常显示。

“Dim ra, rg, rp, Rpa, Rpg, nm, Cpa, Ta As Double”这句代码的意思是定义双精度变量 ra, rg, rp,Rpa, Rpg, nm, Cpa 和 Ta。在 VBA 中可以一次定义多个变量。

最后，应该注意的是：在 VBA 中每句代码的结束不需要增加标点符号，在 VBA 中不能像在 VC++中那样每句代码结束都要加英语中分号。

3 程序计算结果

编制好程序之后，就可以运行程序了。在运行程序时可以选择“逐句读”、“运行到光标处”和“运行子过程/用户窗体”等几个选项。选择“逐句读”时可以按“F8”键，每按一次键程序运行一行。如果选择“运行到光标处”可以按“Ctrl+F8”键，这样就会运行到光标处。如果选择“运行子过程/用户窗体”可以按“F5”键，这样可以一次运行完所有程序代码，但是这样就发现不了程序中存在的错误了。选择“逐句读”的方式可以很快发现程序中的所有错误。

在修改完所有的程序错误之后，就可以选择“运行子过程/用户窗体”直接运行程序了。运行完的程序如图 6 所示。



图 6 计算结果

图 6 中窗体中黑色文本框中的变量均是要输入的变量，蓝色文本框中的数据是计算出来的结果。在输入所有输入变量的具体数值之后，点击“开始计算”按钮，则可以计算出结果了。在输入数据时要注意和文本框后的变量单位相匹配。

这样在编写完了联合循环的性能计算程序之后，就可以随时随地在自己的电脑上计算燃气-蒸汽联合循环发电机组的性能参数了。

4 结语

VBA 程序方便易用，在工程技术领域里面得到了越来越多的应用。在燃气-蒸汽联合循环这种新型的发电领域也得到了应用。为了解燃气-蒸汽联合循环的实时性能参数，可以通

（下转第 40 页）

来卡网出品
LAICAR.COM
shop35833438.taobao.com

Android 手机应用开发之信息平台移动客户端

汪永松

摘　要：从开发者的角度对 Android 手机平台下，信息平台移动客户端较为流行的应用进行了全面的功能分析，并结合 Android 平台的相关技术和应用模式，通过实机案例，详细地介绍了该类应用的开发过程。可用于指导 Android 开发者了解信息平台移动客户端应用的开发模式和实现技巧。

关键词：Android 系统；信息展示；信息发布；移动客户端

1　应用概述

随着移动网络基础设施的建设，手机不可否认地逐渐成为重要的信息载体。不仅如此，随着智能手机在性能和系统功能方面的不断提升和完善，其在信息平台中的地位逐步赶上甚至超过桌面系统。时至今日，对于大多数 Android 手机而言，其不仅能够进行信息的获取和展示，而且也能实现信息的发布。正是这种应用模式的普及，移动客户端的概念应运而生，并且其“光芒”已经逐渐掩盖桌面客户端。而实际上，移动客户端的业务框架是在已有业务框架（B/S）上的扩展，其主要扩展了移动设备通过无线网络到互联网的环节。信息平台服务端—客户端业务框架如图 1 所示。

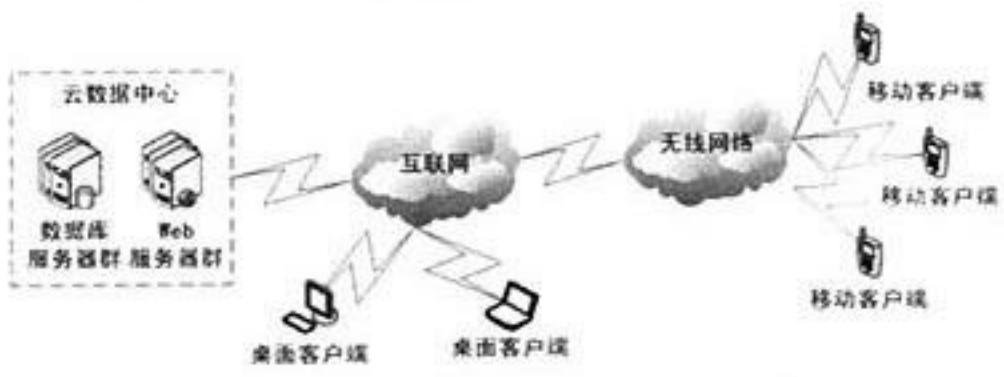


图 1　信息平台服务端—客户端业务框架

图 1 中，无论是桌面客户端（以浏览器为核心）还是移动客户端，其最终都需要通过互联网连接到数据中心，而且客户端一般需要通过 Web 服务器提供的服务接口来进行信息的处理。为了突出重点，文中的信息处理主要包括 3 个方面，即信息的获取、展示和发布。

对于信息的获取，属于后台动作，所以可以考虑两种客户端的统一，即无论是对桌面客户端还是移动客户端，其服务接口均采用统一的传输方式和数据格式；而对于信息的展示和发布，桌面系统和手机系统由于受制于各自性能的影响，其展示方式和发布内容无法做到完全统一，而是需要根据系统的各自特性（例如：用户界面组件、组件间的交互机制等）进行定制。

文中将以 Android 实机（Android 2.3.6）为验证环境，完整地介绍一款分享信息平台移动客户端（如图 2 所示）的开发过程，希望能对 Android 开发者起到参考作用。



图 2　信息平台移动客户端实机界面

2　功能分析

2.1　关键功能及相关技术

对于信息平台移动客户端应用的主要功能，拟从信息的获取、发布和展示 3 个环节进行介绍，对于各环节再从信息的内容和处理方式进行分析。

2.1.1 手机移动客户端信息获取

（1）获取内容

客户端从服务端获取的内容一般以文本和图片居多。其中文本信息都是内容实体，既可以是简单文本，也可以是 HTML 代码，或者是 JSON 格式的结构文本；而图片信息则既可以是内容实体也可以是网页 URL。对于上述两种客户端，其获取内容的主要差异是图片信息。对于桌面客户端而言，无论是对服务端还是本地图片资源，浏览器都会隐式地在后台执行下载操作，统一将图片作为网页缓存进行展示，所以其只需图片的网页 URL 即可；而对于移动客户端，该下载操作必须自己完成，其最终所需的是图片内容实体。

（2）获取方式

在 B/S 业务框架中，客户端通过 HTTP 协议，以异步的方式调用服务端的信息接口，获取返回内容，再对返回内容进行解析，并转换成客户端的展示内容。在桌面客户端中，一般采用

LAICAR.COM
shop35833438.taobao.com

Ajax 技术作为异步调用的机制，以 JSON 作为数据的交互格式。

对于 Android 手机移动客户端，其可以使用异步任务或线程消息队列的机制来实现任务的异步调用；可以使用 org.apache.http 包中的 API 通过 HTTP 的方式发送内容请求；可以使用 org.json 包中的 API 对于返回的 JSON 格式的结构文本进行解析。对于下载操作，也可以使用 HTTP 的方式获取服务端返回的字节流，并保存为本地文件。

2.1.2 手机移动客户端信息发布

（1）发布内容

由于手机设备的特殊性，其发布的信息及途径要比桌面系统丰富得多。对于手机设备而言，其不仅可以发送输入的文本、存储卡中的图片（文件），而且还可以发送即时拍照、手机的位置信息、即时录音、即时视频等。本文中开发案例所发布内容仅包括文本（包括移动客户端标识信息）、图片、即时拍照和位置信息。

（2）发布方式

在 Android 手机平台，文本信息通常采用手工输入，图片信息既可以选择本地文件也可以使用意向（Intent）机制调用相机的拍照功能，而位置信息则可通过系统服务——电话系统管理器（TelephonyManager）的位置 API 来获取，移动客户端标识信息（手机号码或 IMEI）则也需要通过电话信息的 API 来获取。

对于文本信息和图片信息（无论是本地图片还是即时拍照图片），都可以采用 HTTP 的方式进行上传。在 Android 手机移动客户端，可以按照多部分实体（MultipartEntity）的结构将文本实体和文件实体组装在一起，再以 HTTP 的方式将实体内容发送给服务端（实际上该内容与在桌面客户端中通过 Web 页上传表单所发送的内容的结构相同）。

2.1.3 手机移动客户端信息展示

（1）展示内容

移动客户端的展示内容无外乎文本和图片，其中文本内容一般映射于相应的 POJO 类，而图片内容则是本地图片文件（其路径是 POJO 类的属性）。

（2）展示方式

对于单条文本内容，可以使用文本视图组件（TextView）进行展示；对于单项文本列表，则使用微调控制器（Spinner）或自动完成文本框（AutoCompleteTextView）；对于多项文本列表，则使用列表视图（ListView）或扩展列表视图（ExpandableListView）。

对于单个图片可以使用图片视图（ImageView）；对于多个图片则可使用画廊（Gallery）或格子视图（GridView）进行批量展示。

在 Android 平台，按照 MVC 的设计原则，对于多项内容（文本或图片）的展示，都提供了对应的视图（上述的列表视图和画廊视图等）和对应的适配器（Adapter）来管理数据项和控制展示。而开发者要做的是定义数据项和对应的适配器。

2.1.4 基于地理位置的信息分析

对于手机移动客户端而言，地理位置在其周边的信息的重要程度一般都要高于地理位置离其较远的信息，例如：附近的美食、附件的路况等。此类信息的选择需要基于地理位置进行分析，例如：按照与当前手机距离的由近及远来显示分享信息。

在 Android 手机平台，可以使用电话系统的定位功能来获取手机移动客户端的基站位置信息，再通过移动地图服务接口将基站位置信息转换成全球经纬度信息。如此一来，通过判断经纬度的直线距离即可判断信息与本机的距离远近。

2.1.5 系统设置维护

系统设置维护主要用于维护设置信息，设置信息主要包括服务器的主机地址和服务端口。Android 平台提供了首选项（Preference）框架，开发者可以快捷地使用 XML 文件来定义首选项编辑界面。

2.2 功能结构

图 3 是该手机移动应用的功能结构图，其一级功能包括：获取分享信息（最新分享、我的分享和附近分析）、发布分享和系统设置。其中最新分享是按照分享发布时间的新旧程度所选择的所有用户的分享；我的分享是按照分享发布时间的新旧程度所选择的、当前用户所发布的分享；附近分析是按照与当前用户地理位置相隔距离的近远所选择的分享信息。



图 3 信息平台移动客户端功能结构

3 界面设计及接口定义

3.1 界面

3.1.1 主界面

应用程序的主界面如图 2（左）所示，其界面内容主要包含两部分：最新分享列表和选项菜单。主界面的布局定义（XML）如代码 1 所示：

代码 1 分享信息平台移动客户端主界面布局定义

文件名：layout/main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout  xmlns:android = "http://schemas.android.com/
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
apk/res/android"
    android:orientation="vertical"
            android:layout_width = "fill_parent" android:
layout_height="fill_parent">
    <ListView android:id="@id/android:list"
              android:layout_width = "fill_parent" android:
layout_height="fill_parent"
  android:fastScrollEnabled="true" android:focusable="true"
      android:padding="8px"/>
    <TextView android:id = "@id/android:empty" android:
text="无数据"
              android:layout_width = "fill_parent" android:
layout_height="fill_parent"/>
</LinearLayout>
```

主界面的选项菜单定义如代码 2 所示：

代码 2　分享信息平台移动客户端主界面选项菜单定义

文件名：menu/ops_menu.xml

```
<menu xmlns:android = "http://schemas.android.com/apk/res/
android">
    <item android:id = "@ +id/mi_refresh" android:icon = "
@drawable/refresh"
      android:title="最新分享" />
    <item android:icon="@drawable/add" android:id="@+id/
mi_create"
      android:title="发起分享" />
    <item android:id = "@ +id/mi_nearby" android:icon = "
@drawable/nearby"
      android:title="附近分享"/>
    <item android:id = "@ +id/mi_my" android:icon = "
@drawable/my"
      android:title="我的分享"/>
    <item android:id = "@ +id/mi_setting" android:icon = "
@drawable/setting"
      android:title="系统设置"/>
    <item android:id = "@ +id/mi_exit" android:icon = "
@drawable/exit"
      android:title="退出系统"/>
</menu>
```

3.1.2 分享发布界面

分享发布界面如图 2（右）所示，其界面主要内容包含 3 部分：类型选择器、主题输入框、图片选择按钮、地理位置信息栏、发布按钮、图片预览区。限于篇幅，分享发布界面的布局定义请参考附带代码。

3.1.3 分享列表界面

无论是我的分享界面还是附近分享界面，其主要内容还是分享列表，所以其界面布局与主界面的分享列表相同，在此不再赘述。

3.2　数据接口定义

在本案例中，数据内容不仅包括文本还包括图片；不仅需要获取数据，而且还需要上传数据。不仅如此，应用程序还需要调用外部接口来获取手机的经纬度坐标。所以文中的移动应用所涉及的接口包括数据获取和发布接口、图片下载以及地理位置查询接口。

3.2.1 数据获取接口（dataset.ws）

数据获取接口是手机移动客户端获取服务端分享信息的接口，其通过 HTTP 协议，以 JSON 格式（JSON 数组）作为数据交换格式。其输入数据项定义如下：

{ ds_id:<数据集标识>,p1:<参数 1>[,p2:<参数 2>..] }

其输出数据项定义参见数据库设计中【分享信息表】。

3.2.2 数据发布接口（ins.ws）

数据发布接口是手机移动客户端向服务端上传分享信息的接口，其也通过 HTTP 协议，按照 HTTP 实体的格式，将文本内容进行包装，上传至服务端。其输入数据项定义如下：

{ tbl:<数据表>,cols:<插入列名>,vals:<插入列值> }

其输出数据项定义如下：

{ RESULT:<执行结果> }

3.2.3 图片下载接口（download.ws）

图片下载接口用于手机移动客户端从服务端数据库中获取 BLOB 字段中的图片信息并保存到本地存储设备上。其通过 HTTP 协议，以 JSON 格式（JSON 数组）作为请求发送格式，客户端将获取图片内容流。其输入数据项定义如下：

{ tbl:<数据表>,col:<BLOB 字段列名>,pk:<主键列名>,pkval:<主键列值> }

其输出数据项定义如下：

{ RESULT:<执行结果> }

3.2.4 图片上传接口（upload.ws）

图片上传接口用于手机移动客户端将客户端获取的照片（本地照片或相机即时获取的照片）上传并保存到服务端数据库中。其通过 HTTP 协议，以多部分实体的格式，将图片和文本内容进行组合包装，上传至服务器。其输入数据项定义如下：

{ tbl:<数据表>,col:<BLOB 字段列名>,pk:<主键列名>,pkval:<主键列值> }

其输出数据项定义如下：

{ RESULT:<执行结果> }

3.2.5 地理位置查询接口

地理位置查询接口是指手机移动客户端将手机的基站定位信息通过该接口进行查询并获取到对应的全球经纬度信息。其通过 HTTP 协议，以 JSON 格式作为数据交换格式。其接口的地址信息参见代码。

4　数据库设计

4.1　分享信息表

分享信息表主要用于存放移动客户端所发布的分享信息，

LAICAR.COM
shop35833438.taobao.com

其内容如表 1 所示。

表 1 分享信息表

表名：TAB_SHARING			
列名	类型	约束	备注
S_ID	VARCHAR2 (32)	PK	分享标识
S_TYPE	VARCHAR2 (32)	FK	分享类型
S_TITLE	VARCHAR2 (100)	NOT NULL	标题
S_USER	VARCHAR2 (32)	NOT NULL	发布用户
S_TSP	TIMESTAMP	NOT NULL	发布时间戳
S_PIC	BLOB		附带图片
S_LAT	INTEGER	NOT NULL	发布时位置纬度（乘以 106）
S_LON	INTEGER	NOT NULL	发布时位置经度（乘以 106）

4.2 数据集定义表

数据集定义表主要用于存放移动客户端所需的数据集的定义信息，其内容如表 2 所示。

表 2 数据集定义表

表名：SYS_DATA_SET			
列名	类型	约束	备注
DS_ID	VARCHAR2 (32)	PK	数据集标识
DS_TITLE	VARCHAR2 (100)	NOT NULL	标题
DS_TYPE	VARCHAR2 (32)	NOT NULL	类型（1：SQL，2：存储过程）
DS_DS	VARCHAR2 (1000)	NOT NULL	数据集（SQL或存储过程）
DISPNO	INTEGER		显示顺序

所谓数据集的定义即预先将获取指定数据集的 SQL 语句或存储过程写好并存入到数据库。客户端通过数据获取接口调用指定的数据集并传入参数，服务端引擎负责执行该数据集所写的 SQL 语句或存储过程，并将结果以 JSON 的格式返回给客户端。

5 程序框架及功能实现

5.1 程序框架

为了更好地阐明该程序的组成结构，作者将程序所包含的组件按照功能特性归集为 4 个部分：服务端、支撑层、数据层和展示层，后 3 层均属于移动客户端。其中，服务端主要是提供分享信息接口；支撑层包含：位置服务、位置查询线程、（图片）文件下载线程和信息操作模式工具类，支撑层的分享信息获取工具类调用服务端的接口来获取和上传分享乐信息；数据层包括分享信息列表、线程消息队列和列表适配器（包装了分享信息列表）；展示层主要是 Activity 组件，其中主 Activity 负责调用我的分享 Activity、附近分享 Activity、发布分享 Activity 和设置管理 Activity。

虽然各个组件被划分到不同的层，然后展示层、数据层和支撑层中的各个组件彼此都存在紧密的关联。展示层中的主 Activity、我的分享 Activity 和附近分享 Activity 中的列表视图都需要绑定分享信息列表适配器；文件下载线程（在列表记录类中调用）通过向主 Activity 的线程消息队列发送消息来通知图片下载完毕；共享信息操作模式工具类（由分享列表管理器调用）通过向各个 Activity 的线程消息队列发送消息来通知数据获取完毕。各层组件间的关联关系如图 4 所示。



图 4 程序组件关联关系

5.2 主要功能实现

5.2.1 从服务端获取分享信息

（1）获取分享文本

实际上，主 Activity、附近分享 Activity 和我的分享 Activity 都存在获取分享信息，本文以主 Activity 为例进行介绍。代码 3 是主 Activity 初始化代码：

代码 3 主 Activity 初始化代码

文件名：MainAct.java

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    //初始化线程消息队列处理器
    handler = new Handler() { … };
    //初始化应用程序文件夹
    FooPrefsUtil.getInstance().initAppDir();
    //启动定位服务先
    LocationService.startLocService(this);
    //获取共享列表
    SharingMgr.getInstance().updLatestList(this);
    //设置列表视图适配器
    mListAdapter = new FooListAdapter(this, latestSharing,
            R.layout.row, new int[] { R.id.tv_type, R.id.tv_title,
R.id.tv_user,
            R.id.tv_tsp, R.id.tv_location, R.id.iv_pic }
    );
    setListAdapter(mListAdapter);
    //设置分隔
    getListView ().setDivider (this.getResources ().getDrawable
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
(R.drawable.green));
    getListView().setDividerHeight(10);
}
```

代码 3 中，主 Activity 先通过共享列表管理器的方法获取最新的分享信息，再将数据集与列表视图适配器进行绑定。代码 4 是获取最新分享信息的关键代码：

代码 4　获取最新分享信息

文件名：SharingMgr.java

```
public void updLatestList(Context ctx) {
    //填充参数
    Hashtable <String,String > params = new Hashtable < String,String>();
    params.put("ds_id", "sharing_latest_sel");
    //更新服务接口的 URL
    final String url = FooPrefsUtil.getInstance ().updateUrl (ctx, IDef.Url_Model_Read);
    //启动数据请求线程
    HttpReqThread t = new HttpReqThread (ctx, url, params, IDef.Flag_Latest);
    t.start();
}
```

代码 4 中，使用 HTTP 请求线程发出数据请求，代码 5 是该线程的执行代码：

代码 5　HTTP 请求线程执行代码

文件名：HttpReqThread.java

```
@Override
public void run() {
    JSONArray arr = FooHttpUtil.getInstance ().doPost2(url, params);
    if(arr==null) { return; }
    switch(typeFlag) {
    case IDef.Flag_Latest: { getLatest(this.ctx, arr); break; }
        case IDef.Flag_My: { getMy(this.ctx, arr); break; }
    case IDef.Flag_Nearby: { getNearby(this.ctx, arr); break; }
        default: { break; }
    }
}
```

代码 5 中，HTTP 请求线程先发送请求并获取记录集（JSON 数组），再依据请求主体的类型标识的不同而更新不同的数据集（最新的、当前用户的、当前用户附件的）。代码 6 是发送 HTTP 请求并获取记录集（JSON 数组）的主要代码：

代码 6　发送 HTTP 请求获取记录集

文件名：FooHttpUtil.java

```
public JSONArray doPost2 (String url, Map<String, String> params) {
    //初始化字符串缓冲(用于拼凑记录集)
    StringBuffer sb = new StringBuffer();
    try {
        // 创建连接,发送请求并接受回应
        DefaultHttpClient client = new DefaultHttpClient();
        HttpPost post = new HttpPost(url);
        //包装请求参数
        if (params ! = null && params.size() > 0) {
    List <BasicNameValuePair > pairs = new ArrayList < BasicNameValuePair>(params.size());
            Set<String> keys = params.keySet();
            for (String key : keys) {
  pairs.add(new BasicNameValuePair(key, params.get(key)));
            }
 post.setEntity(new UrlEncodedFormEntity(pairs, "utf-8"));
        }
        HttpResponse resp = client.execute(post);
        //返回状态
    int statusCode = resp.getStatusLine().getStatusCode();
        if (statusCode == HttpStatus.SC_OK) {
            //获取回应条目(JSON 格式)
            HttpEntity entity = resp.getEntity();
            if(entity ! = null) {
                InputStream is = entity.getContent();
            BufferedReader br = new BufferedReader (new InputStreamReader(is));
                //获取服务端返回内容
                String line = null;
                while ((line=br.readLine())! =null) {
                    sb.append(line);
                }
                br.close();
            }
        }
        post.abort();
        //构建 JSON 数组
        JSONArray arr = new JSONArray(sb.toString());
        return (arr);
    } catch( ClientProtocolException e) { e.printStackTrace();
    } catch (IOException e) { e.printStackTrace();
    } catch(JSONException e) { e.printStackTrace();
    }
    return (null);
}
```

代码 6 中，使用 Apache 提供的 HTTP 客户端工具类进行请求的发送和接收。代码 7 是代码 5 中 HTTP 请求线程更新最新分享信息的关键代码：

代码 7　更新最新分享信息

文件名：HttpReqThread.java

```
private void getLatest(Context ctx, JSONArray arr) {
    //POJO 对象数组
    ArrayList <Sharing > ds = new ArrayList <Sharing >(arr.length());
    for(int i = 0; i < arr.length(); ++i) {
        JSONObject obj = null;
```


LAICAR.COM
shop35833438.taobao.com

```
        try {
            obj = (JSONObject)(arr.get(i));
        } catch(JSONException e) {
            e.printStackTrace();
        }
        //将 JSON 对象转换成 Pojo 类
    Sharing sharing = Json2Pojo(ctx, obj, IDef.Flag_Latest);
        if(sharing ! = null) {
            Log.d(IDef.App_Tag, sharing.toString() );
                ds.add(sharing);
            }
        }
        //更新主 Activity 的数据集
        MainAct.updData(ds);
        //通知主 Activity 的数据集更新
        Message msg = new Message();
        msg.what = IDef.Msg_Data_Upd;
        MainAct.postMessage(msg);
}
```

代码 7 中，需要现将 JSON 对象转换成 POJO 类对象，然后用该类数组来更新对应 Activity 的数据集（此处是主 Activity）并向该 Activity 发送数据存在更新的消息。代码 8 是将 JSON 对象转换成 POJO 类的主要代码：

代码 8　将 JSON 对象转换成 POJO 类

文件名：HttpReqThread.java

```
//JSON 对象转换成 POJO(Plain Old Java Objects)类
private Sharing Json2Pojo (Context ctx, JSONObject obj, int
typeFlag) {
    if(obj==null) { return (null); }
    try {
        final String id = obj.getString("S_ID");
        final String type = obj.getString("S_TYPE");
        final String title = obj.getString("S_TITLE");
        final String user = obj.getString("S_USER");
        final String tsp = obj.getString("S_TSP");
        final int lat = obj.getInt("S_LAT");
        final int lon = obj.getInt("S_LON");
        //添加到列表中
        Sharing jobj = new Sharing (ctx, id, type,title,user,
tsp,lat,lon, typeFlag);
        return (jobj);
    } catch(JSONException e) { e.printStackTrace();
    }
    return(null);
}
```

（2）获取图片

分享信息的图片信息存放在 BLOB 字段中，需要通过下载接口去获取并保存为本地文件。代码 9 是分享信息 POJO 类构造函数，图片下载在其中调用：

代码 9　分享信息 POJO 类构造函数

文件名：Sharing.java

```
public Sharing (Context ctx, String id, String type,String title,
String user, String tsp,
            int lat, int lon,int typeFlag) {
    this.id = id;
    this.type = type;
    this.title = title;
    this.user = user;
    this.tsp = tsp;
    this.lat = lat;
    this.lon = lon;
his.picFilePath = FooPrefsUtil.getInstance().getDownloadDir()
+File.separator+id+".jpg";
    //下载图片信息
DownloadThread t = new DownloadThread(ctx,this,typeFlag);
    t.start();
}
```

代码 9 中，分享信息类通过调用下载线程来启动对分享图片的下载。代码 10 是图片文件下载线程的执行代码：

代码 10　图片文件下载线程执行代码

文件名：DownloadThread.java

```
@Override
public void run() {
    //获取分享记录 ID
    final String id = sharing.getId();
    //获取图片保存路径
    File dir = new File(destDir);
    if(! dir.exists()) {
        return;
    }
File destFile = new File(destDir+File.separator+id+".jpg");
    if(destFile.exists()) {
        return;
    }
    //发送下载请求
    Hashtable <String,String > params =  new Hashtable <
String,String>();
    params.put("tbl", "TAB_SHARING");
    params.put("col", "S_PIC");
    params.put("pk", "S_ID");
    params.put("pkval", id);
    //更新下载接口 URL
    final String url = FooPrefsUtil.getInstance ().updateUrl
(ctx, IDef.Url_Model_Download);
    //调用 Web Service 下载图片
    boolean result = FooHttpUtil.getInstance().download(url,
params, destFile);
    if(result==true) {
    this.sharing.setPicFilePath(destFile.getAbsolutePath() );
        Log.d (IDef.App_Tag, " 下载图片 【" +destFile.
getAbsolutePath()+"】完成。");
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
        switch(this.typeFlag) {
        case IDef.Flag_Latest: { //向主 Activity 发送消息
                    Message msg = new Message();
                    msg.what = IDef.Msg_Img_Upd;
                    MainAct.postMessage(msg);
                    break;
            }
    case IDef.Flag_My: { //向我的分享 Activity 发送消息
                    …
                }
                        case IDef.Flag_Nearby: { //像附近分享
//Activity 发送消息
                            …
                }
                default: { break;
                }
            }
        }
}
```

代码 10 中，图片文件下载线程通过 HTTP 工具类发起文件下载请求，下载完毕后向各数据展示 Activity 发送消息，通知其更新数据集。限于篇幅，HTTP 类中文件下载的处理请参考附带代码。

(3) 对数据获取完毕的响应

在代码 10 和代码 7 中，当分享信息的数据获取完毕后，执行线程都会向相应的 Activity 发送消息。那么作为消息接收方的 Activity 又是如何响应的呢？代码 11 是主 Activity 中初始化线程消息处理器的主要代码：

代码 11　主 Activity 中初始化线程消息处理器

文件名：MainAct.java

```
//初始化线程消息队列处理器
handler = new Handler() {
    @Override
    public void handleMessage(Message msg) {
        switch(msg.what) {
            case IDef.Msg_Data_Upd:  { mListAdapter.
notifyDataSetChanged(); break;
            }
            case IDef.Msg_Img_Upd:  { mListAdapter.
notifyDataSetChanged(); break;
            }
            case IDef.Msg_Loc_Upd: {
        Toast.makeText (MainAct.this, " 地址获取成功！ ",
Toast.LENGTH_LONG).show();
                break;
            }
    case IDef.Msg_Loc_Service: { initLocation(); break;
            }
            default: { break; }
        }
    }
};
```

代码 11 中，主 Activity 使用线程消息处理器来接收消息，并按照消息类型进行不同的处理。当收到数据存在更新和图片存在更新的消息后的处理是通知列表适配器更新数据集；收到位置服务启动完毕的消息后的处理时初始化位置信息。

5.2.2 发布分享信息到服务端

(1) 发布文本信息

代码 12 是发布分享信息的主要代码：

代码 12　发布分享信息

文件名：PushAct.java

```
private void pushSharing() {
    String title = titles.getText().toString().trim();
    //发布内容的有效性检查
    if((title==null)||(title! =null&&title.length()<2) ) {
        Toast.makeText (this, " 主题不能为空！ ", Toast.
LENGTH_SHORT).show();
        return;
    }
    Location loc = MainAct.getLocation();
    //数据准备
    final String cols = "S_ID,S_TYPE,S_TITLE,S_USER,
S_TSP,S_LAT,S_LON";
    final String tid =  ((id==null)?FooSysUtil.getInstance().
getIdOfTsp():id);
    final SharingType type = (SharingType)types.
getSelectedItem();
    final String user = MainAct.getUserId();
    final String tsp = FooSysUtil.getInstance().getTsp();
    final String location = ""+loc.getLat()+","+loc.getLon();
    final String vals = "" +tid + "," +type.getId () + "," +
URLEncoder.encode(title)+"","+
     user+"",to_date('"+URLEncoder.encode(tsp)+"','"+
     URLEncoder.encode("yyyy-mm-dd hh24:mi:ss")+"'),"+
location;
    //包装参数
    Hashtable<String,String> params = new Hashtable<
String,String>();
    params.put("tbl", "TAB_SHARING");
    params.put("cols", cols);
    //更新发布接口的 URL
final String url = FooPrefsUtil.getInstance().updateUrl(this,
        IDef.Url_Model_Create+"?vals="+vals);
    Log.d(IDef.App_Tag, url);
    //调用信息发布接口进行发布
   JSONObject obj = FooHttpUtil.getInstance ().doPost(url,
params);
    if(obj == null) { return; }
    try { //若文本内容发送成功，则即可发布图片
        String result = obj.getString(IDataSpec.RESULT);
```


shop35833438.taobao.com

```
        if(result.compareTo(IDataSpec.RESULT_OK)==0) {
            //上传图片
            uploadPic(tid);
            //通知主 Activity 更新
            Message msg = new Message();
            msg.what = IDef.Msg_Data_Upd;
            MainAct.postMessage(msg);
            Toast.makeText(this, "分享发布成功！", Toast.
LENGTH_LONG).show();
            finish();
        }
    } catch(JSONException e) { e.printStackTrace();
    }
}
```

代码 12 中，其思路是先获取发布信息内容（分享标识、类型、用户、时间戳等），再包装成接口参数，最后调用服务端发布接口进行信息的发布。限于篇幅，调用服务端发布接口的代码请参考附带代码。

（2）发布图片信息

若文本信息发布成功则即可发布图片信息，代码 13 是代码 12 中发布图片的主要代码：

代码 13　发布图片信息

文件名：PushAct.java

```
private void uploadPic(String id) {
    if(picFilePath==null) { return; }
    //包装参数
    Hashtable<String,String> params = new Hashtable<
String,String>();
    params.put("tbl", "TAB_SHARING");
    params.put("col", "S_PIC");
    params.put("pk", "S_ID");
    params.put("pkval", id);
    //
    File f = new File(picFilePath);
    final String url = FooPrefsUtil.getInstance().updateUrl
(this, IDef.Url_Model_Upload);
    //调用 Web Service 上传图片
    JSONObject obj = FooHttpUtil.getInstance().upload(url,
params, f);
    if(obj == null) {
        Toast.makeText(this, "图片发送失败！", Toast.
LENGTH_SHORT).show();
        return;
    }
    try {
        String result = obj.getString(IDataSpec.RESULT);
        if(result.compareTo(IDataSpec.RESULT_OK)==0) {
            //Toast.makeText(this, result, Toast.
LENGTH_SHORT).show();
        }
    } catch(JSONException e) {
        e.printStackTrace();
        Toast.makeText(this, "图片发送失败！", Toast.
LENGTH_SHORT).show();
    }
}
```

代码 13 中，共享发布 Activity 通过调用图片上传接口完成图片信息的发布。

（3）对信息发布完毕的响应

在代码 12 中，当分享信息发布完毕后即向主 Activity 发送了一条数据存在更新的消息，用以通知主 Activity 更新其数据集。

5.2.3 移动客户端展示分享信息

分享信息的展示可以考虑从分享信息列表视图的行布局（对应列表视图中的每一行的布局）和分享信息列表适配器（将数据显示到布局中）。

（1）分享信息列表视图的行布局

分享信息列表视图的行视图效果如图 2（左）中的每一项，限于篇幅，分享信息列表视图的行布局定义请参考附带代码。

（2）分享信息列表适配器

代码 14 是分享信息列表适配器的关键代码，其中主要是如何更新其行布局：

代码 14　分享信息列表适配器

文件名：FooListAdapter.java

```
@Override
public View getView (int pos, View convertView, ViewGroup
parent) {
    if(convertView == null) {
    final LayoutInflater inflater = ctx.getLayoutInflater();
        //获取项目 View(每一行)
        convertView = inflater.inflate(layoutId, null, false);
    }
    //按照行属性设置其背景
    if(pos == selectedPos) {
 convertView.setBackgroundResource(R.drawable.lorange);
    } else {
        if((pos%2)>0) { //奇数
    convertView.setBackgroundResource(R.drawable.gray);
        } else { //偶数
    convertView.setBackgroundResource(R.drawable.silver);
        }
    }
    if(items! =null) { //将分享对象的属性信息展示到行布局
//所定义的组件中
        final Sharing item = items.get(pos);
        ((TextView)convertView.findViewById (this.to [0])).
setText(item.getType());
        ((TextView)convertView.findViewById (this.to [1])).
setText(item.getTitle());
        ((TextView)convertView.findViewById (this.to [2])).
setText(item.getUser());
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
        ((TextView)convertView.findViewById (this.to[3])).
setText(item.getTsp());
        final String location = "("+item.getLatF ()+","+item.
getLonF()+")";
        ((TextView)convertView.findViewById (this.to[4])).
setText(location);
        //显示分享图片的缩略图
                final int width = (int)ctx.getResources().
getDimension(R.dimen.img_width);
                final int height = (int)ctx.getResources
().getDimension(R.dimen.img_height);
        final Bitmap bitmap =
            FooPicUtil.getInstance ().getLocalBitmap (item.
getPicFilePath(), width, height);
        ((ImageView)convertView.findViewById(this.to[5])).
setImageBitmap(bitmap);
        //Log.d(IDef.App_Tag, item.toString());
    }
    return (convertView);
}
```

代码 14 中，适配器依据回调函数中记录行位置参数（pos）获取对应的记录对象，然后获取记录对象的属性信息并设置给行布局中的显示组件。

5.2.4 移动客户端获取本地信息

在移动客户端所发布的信息中，相机即时拍照所获取的照片以及手机设备的位置和标识信息，这些都是在桌面客户端中无法获取的内容，而正是这些手机平台特有的信息，让移动客户端的应用变得丰富多彩。

（1）获取即时拍照照片

代码 15 是调用系统相机拍摄分享照片的主要代码：

代码 15　调用系统相机拍摄分享照片

文件名：PushAct.java

```
private void takePic() {
  id = FooSysUtil.getInstance().getIdOfTsp();
        final String filePath = FooPrefsUtil.getInstance().
getUploadDir()+File.separator+id+".jpg";
            Intent i = new Intent (MediaStore.
ACTION_IMAGE_CAPTURE);
  i.putExtra (MediaStore.EXTRA_OUTPUT, Uri.fromFile(new
File(filePath)));
  this.startActivityForResult(i, IDef.Req_Code_Camera);
}
```

代码 15 中，通过指定输出路径的方式调用系统相机，当调用成功，则指定路径即为拍照照片的路径。对于使用不指定输出路径的方式调用系统相机，在回调函数中将直接获取照片的位图信息，而且该位图不是实际照片的内容，而是缩略图。

（2）获取设备标识及位置信息

对于手机设备标识及位置信息，需要使用电话系统服务（TelephonyManager）的相关接口，代码 16 是获取设备标识及位置信息的主要代码：

代码 16　获取设备标识及位置信息

文件名：LocationService.java

```
public void updateLocation() {
    if(telMgr==null) { return; }
    CellLocation loc = telMgr.getCellLocation();
    if(loc==null) { return; }
    updateLocation(loc);
}
public void updateLocation(CellLocation loc) {
    if(loc==null) { return; }
    //获取用户标识(电话号码或 IMEI)
    userId = telMgr.getLine1Number();
    if(userId==null) { userId = telMgr.getDeviceId(); }
    if(userId==null) { return; }
    //获取手机当前的位置区域识别信息
final String operator = this.telMgr.getNetworkOperator();
    final int mcc = Integer.parseInt(operator.substring(0,3));
    final int mnc = Integer.parseInt(operator.substring(3));
    final int lac = ((GsmCellLocation)loc).getLac();
    final int cid = ((GsmCellLocation)loc).getCid();
    //判断定位信息合法性
    if((lac==0)&&(cid==0)) { return; }
if( (lac < 0) || (lac >= 0xffff) || (cid < 0) || (cid >= 0xffff) ) {
        return;
    }
    lai.setMcc (mcc); lai.setMnc (mnc); lai.setLac (lac); lai.
setCid(cid);
}
```

代码 16 中，使用系统电话管理器的相关 API 可以获取位置信息（基站位置信息）、手机号码、设备 IMEI 和网络标识。需要注意的是当前的位置信息还仅仅是手机网络的基站位置信息，还需要调用互联网服务接口通过基站位置信息获取全球经纬度信息。限于篇幅，该互联网服务接口的调用过程请参考附带代码。

6　结语

对该分享信息平台移动客户端的部分功能细节，并没有给予详细的介绍（例如：通过电话基站位置信息获取全球经纬度信息、获取分享类型信息、上传及下载图片的细节控制、获取附近分享信息等），读者可以通过配套代码进行理解。

该开发实例涵盖了移动客户端应用的常见模式，例如：数据的获取和发布、图片的下载及上传、获取手机位置信息、调用系统相机拍照、基于地理位置的信息关联分析等。在数据的获取过程中，详细介绍了使用线程消息队列机制实现数据获取的异步操作，这些技巧都是在商业项目开发中常用的方式。

该开发案例经 Android 2.3.6 和 4.0.4 版本的实机验证，希望对读者有所参考。

（收稿日期：2013-03-29）

LAICAR.COM
shop35833438.taobao.com

最优数字分配策略的分析与实现之一

——实现思想

尉鹏博　韩银锋

摘　要： 最优数字分配策略是将 1–30 的整数存入 50*50 的矩阵中每个存储单元的 2–5 个子单元中，使得分配的数据满足违约分最低的项目，项目根据不同的方案综合实现违约分最低。

关键词： 最优数字分配；矩阵；相邻违约；相邻的相邻违约

1　题目背景说明

假设有 2500 个数据存储单元，形成为 50*50 的正方形矩阵。每个数据存储单元允许存储 2~5 个整数，整数范围为 1~30，每个整数使用次数不限。

50*50 数据存储单元如表 1 所示。

表 1　50*50 矩阵

1–1	1–2	1–3	1–4	…
2–1	2–2	2–3	2–4	…
…	…	…	…	…

其中 1–1、1–2 等表示存储单元在矩阵中的行列位置，也可以将其作为存储单元的名称。

要求：

（1）将 1~30 的整数按各存储单元的整数个数要求，分配到各数据存储单元中。

（2）每个存储单元内的整数不能相同且不能相邻。如：1 和 2、2 和 2、2 和 3 不能出现在同一个存储单元中。如果违反规则，按下述罚则处理：针对每个存储单元都统计一遍，出现一次相同，结果累加 100 违约分；出现一次相邻，结果累加 50 违约分。

（3）每个存储单元与相邻的存储单元内的整数不能相同且不能相邻。如：第 2–2 存储单元内的整数与第 1–1、1–2、1–3、2–1、2–3、3–1、3–2、3–3 存储单元内的整数不能相同且不能相邻。如果违反规则，按下述罚则处理：针对每个存储单元都统计一遍，出现一次相同，结果累加 20 违约分；出现一次相邻，结果累加 10 违约分。

（4）每个存储单元与相邻存储单元的相邻存储单元内的整数不能相同。如：第 2–2 存储单元内的整数与第 1–4、2–4、3–4、4–1、4–2、4–3、4–4 存储单元内的整数不能相同。如果违反规则，按下述罚则处理：针对每个存储单元都统计一遍，出现一次相同，结果累加 1 违约分。

违约分计算举例，如表 2 所示。

表 2　分配结果表

行\列	列 1	列 2	…
行 1	1,3,5	7,9	…
行 2	11,12,13	1,4	…
…	…	…	…

表 1 所示为部分存储单元内整数分配结果，违约值计算方法如表 3 所示。

表 3　违约值表

当前存储单元标识	冲突存储单元标识	当前存储单元违规整数	冲突存储单元违规整数	违约分数
1–1	2–2	1	1	20
1–1	2–2	3	4	10
1–1	2–2	5	4	10
2–2	1–1	1	1	20
2–2	1–1	4	3	10
2–2	1–1	4	5	10
2–1	2–1	11	12	50
2–1	2–1	12	13	50

为了方便计算，同存储单元内（如 2–1）的冲突整数按照规则计算一次违约分，相邻存储单元内（如 1–1 和 2–2）的冲突整数由于参照主体改变会重复计算违约分。最后的总分为所有违约分的总和。

本题目的业务背景：在移动通信过程中，移动用户的通信内容通过移动台（手机等移动通信设备）转换成频率信号后和通信基站（专门负责收发无线信号的物理通信设备）之间进行交互传输。

国家相关部门为各电信运营商划定专门的频率区间，此频率区间按照相关电信标准可平均划分若干等份小的频率区间作为信号传输的最小单位，每个最小单位最多可同时为 8 个移动用户服务。各电信运营商频率区间不一样，频率带宽也不一样，即可划分的最小单位个数也不一样。

来卡网出品
LAICAR.COM
shop35833438.taobao.com

因此，各电信运营商需要根据用户密度在有限的频率资源条件下，建立多个通信基站，每个通信基站分配 1 个或多个服务频率，以服务通信区域内的用户群。同时，如果基站内或邻近的基站使用相同或相近的频率收发信号，这些基站由于距离位置关系，会相互产生不同程度的电磁干扰，导致基站服务的移动用户出现不同程度的通信障碍。

因此，各电信运营商在建立基站并分配服务频率的过程中会充分考虑周围基站的服务频率，使各基站的服务频率要尽量不与周围基站的服务频率相近，以减少通话过程中的干扰问题。在没有更好的频率资源可用的情况下，运营商会考虑重新分配所有基站的服务频率，达到降低干扰的目的。

根据业务背景，可按如下所述理解题目条件：50*50 的存储单元矩阵为某电信运营商服务某区域的 50*50 个通信基站。1~30 的整数为某电信运营商的可用频率资源。每个存储单元内的整数个数为基站内服务频率个数。2~5 可理解为同基站内或相邻基站内尽量不能出现频率相近的情况，如果出现这种情况，不同距离位置的基站会产生不同程度的通话干扰。

2 问题分析

(1) 首先，考虑自身单元格内的整数不能相同且不能相邻，如果违反规则，针对每个存储单元都统计一遍，出现一次相同，结果累加 100 违约分；出现一次相邻，结果累加 50 违约分。因为自身单元格违约后产生的违约分值很大，所以在分配数据存储的时候直接剔除自身单元格违约的情况。

(2) 考虑每个存储单元与相邻的存储单元内的整数不能相同且不能相邻；如果违反规则，针对每个存储单元都统计一遍，出现一次相同，结果累加 20 违约分；出现一次相邻，结果累加 10 违约分。

(3) 考虑每个存储单元与相邻存储单元的相邻存储单元内的整数不能相同，如果违反规则，针对每个存储单元都统计一遍，出现一次相同，结果累加 1 违约分。

(4) 针对 2、3 的条件，在分配 1–30 个整数存储到单元格内时，对于“相邻违约”和“相邻的相邻违约”两种情况，首先，分配整数进行存储时先取与相邻单元格内整数违约个数最小的整数。其次，利用这些相邻违约个数最小的整数，对相邻的相邻单元格进行判断选取违约个数最小的整数，则存入该整数于单元格内。

(5) 因为 50*50 数据单元格中存储 1~30 整数的个数存储在 Excel 文件中，并将最终分配好的 50*50 的矩阵存入另一 Excel 文件中。所以本项目先要实现读写 Excel 工作表中各数据。

3 系统实现流程

系统实现的流程如图 1 所示。

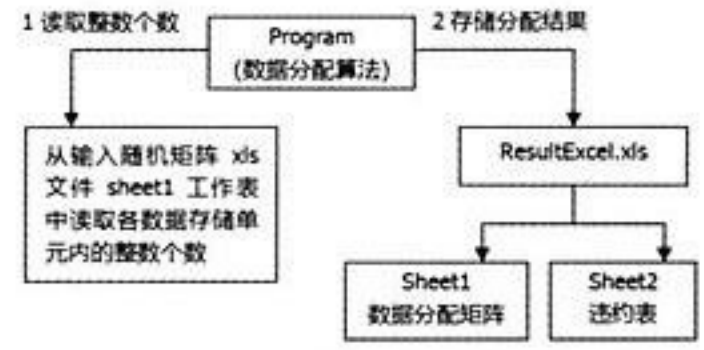


图 1 系统实现流程图

4 项目准备

4.1 实现思想

(1) 对数据分配后的存储问题进行了分析，因为存储数据需对应相应的单元格及单元给定存储的整数个数，所以采用三维整型数组存储。

(2) 在数据的分配问题上因为当前单元格内违约产生的分值很大，则在分配时直接避免当前单元格内的违约。对于“相邻违约”，定义一个 Vaild 实体类，来进行判断取值。因为“相邻违约”所产生的违约分值较大，所以取出“相邻违约”个数最小的值。与此同时定义一个 VaildNear 实体类，以“相邻违约”个数最小的值为基础，进行判断取出与之对应的“相邻的相邻违约”个数最小的值后将该值存入单元格内，在填充存储数据时采用顺序、S 型、水波纹、反水波纹等填充方法来填充，使违约分达到最低。

(3) 数据分配算法读取输入随机矩阵 Excel 文件中 Sheet1 工作表中各数据存储单元内的整数个数，在规定时间内计算产生结果，结果存储在 ResultExcel.xls 文件 Sheet1 表中，同时将违反规则的详细数据存储在 ResultExcel.xls 文件 Sheet2 表中。

4.2 使用 C# 实现

C# 对 Excel 工作表中单元格数据进行读写。在项目中添加文件夹 dao，dao 文件夹中添加类 AccessExcel。添加引用 com 中的 Microsoft Excel 11.0 Object Liberary。类代码如下：

```
public class AccessExcel
  {  private Application excel = null;//Excel 对象
     private Workbook workbook1 = null;//工作簿
     private Worksheet worksheet1 = null;//工作表
  /// 打开 XLS 文件,返回工作簿
public void openXls(string path)
     {  excel = new Application();//新建 excel 对象
        workbook1 = excel.Workbooks.Open (path, Type.
Missing, Type.Missing, Type.Missing, Type.Missing, Type.
Missing, Type.Missing, Type.Missing, Type.Missing, Type.
Missing, Type.Missing, Type.Missing, Type.Missing, Type.
Missing, Type.Missing);//打开工作簿
     }
     ///读 xls 工作表中内容
publicint[,] readStrXls()
     {int[,] arr = new int[50, 50];//实例化二维 int 数组
int i, j;//二维数组下标,i 行,j 列
```

```
        worksheet1 = workbook1.Worksheets[1];//生成工作
//簿中的 1 工作表
for (i = 2; i <= 51; i++)
{  for (j = 2; j <= 51; j++)
{ Range range = worksheet1.Cells[i, j];//设置范围 i 行 j 列单元格
arr[i - 2, j - 2] = Convert.ToInt16(range.Value);
                        //取范围的值转成 int 类型存入数组中
            }
        }
returnarr;
      }
      /// 关闭 Excel
public void closeXls()
      {   worksheet1 = null;
          workbook1 = null;
excel.Quit();
                    System.Runtime.InteropServices.Marshal.
ReleaseComObject(excel);//释放对象
excel = null;
System.GC.Collect();//收集内存垃圾
      }
      /// 将 int 三维数组转换为 string 二维数组,
          ///arr:整型二维数组 51*51,0 行、0 列未用,从第 1 行,
//第 1 列开始取
public string[,] threeString(int[, ,] arr)
      {   string[,] strarr = new string[50, 50];
int i, j, k;//i 为三维数组的行,j 为三维数组的列,k 为元素的下标
for (i = 0; i < 50; i++)
{  for (j = 0; j < 50; j++)
              {   string str = "";//将元素变成 string 类型。
for (k = 0; k < 5 &&arr[i + 1, j + 1, k] ! = -1; k++)
{ if (str ! = "") str = str + ",";//数字间","隔开,第 1 个","不加
str = str + arr[i + 1, j + 1, k].ToString();
                  }
strarr[i, j] = str;//存元素
              }
          }
returnstrarr;
      }
    }
```

4.3 初始化 50*50 存储单元

在项目 dao 文件夹中添加类 DistributeCount，类中添加变量，添加方法 initMatrix。初始化矩阵 51*51，0 行、0 列未用，从第 1 行、第 1 列开始存数据，均为-1，目的是从 1 开始存数据。如果初始是 0，避开相邻违约就要从 2 开始。

（1）变量：

```
int[, ,] a = new int[51, 51, 5];
//50*50 矩阵,每个元素有 2-5 个整数,0 行、0 列未用,从 1
//行、1 列存数据
string[,] strvalid = new string[65536, 5];
//二维字符串数组,记录违约,0 当前单元格位置,1 冲突单元格位
//置,2 当前单元格违规整数,3 冲突单元格违规整数,4违约分
int total = 0;//违约总分
int last = 0;//违约数组记录个数
Valid[] validarr = new Valid[30];
 //存相邻违约值 1-30,及违约个数,记违约位置(下标 i-j-
//p),按违约个数排序
ValidNear[] validneararr = new ValidNear[31];
//存相邻的相邻违约值 0,1-30,的违约个数
string[,] validabsolut = new string[20, 2];
//存相邻违约位置(下标 i-j-p)。1:相邻违约相邻,2:相邻违约相等
string[] validnearabsolut = new string[40];
//存相邻的相邻违约位置(下标 i-j-p)
Valid varvalid = new Valid();
//相邻违约实例,存当前比较值 1-30 的相邻违约信息
ValidNearvarvalidnear = new ValidNear();
 //相邻的相邻违约实例,存当前比较值 1-30 的相邻的相邻
//违约信息
```

（2）初始化 50*50 矩阵方法：

```
public void initMatrix()
{  int i, j, k;
   for (i = 0; i <= 50; i++)//行
   { for (j = 0; j <= 50; j++)//列
{  for (k = 0; k < 5; k++)//5 个元素
{ a[i, j, k] = -1; }
}
}
}
```

参考文献

[1] Nick Randolph.VisualStudio2010 高级编程 [M]．清华大学出版社，2012，01.

[2] 邱仲潘．Visual Basic 2010 (中文版) 从入门到精通 [M]．电子工业出版社，2011，01.

[3] 汪沁．数据结构 [M]．清华大学出版社，2009，09.

（收稿日期：2013-04-02）

（上接第 28 页）

过 Excel 编写这样的计算程序。编写了这样的计算程序之后就可以随时随地了解当前机组的运行情况，从而及时针对机组的运行偏差提出解决措施。

参考文献

[1] 魏汪洋．从零开始学 Excel VBA/[M]．北京：电子工业出版社，2011.

[2] 广东电网公司电力科学研究院 & 华北电力大学．ASME PTC 47-2006 整体煤气化联合循环电站性能试验规程 [S]．北京：中国电力出版社.

（收稿日期：2013-04-12）

来卡网出品 LAICAR.COM shop35833438.taobao.com

基于 Excel VBA 的会计单据演示训练系统开发（三）

何 燕

摘 要：在财经类专业的会计学课程教学中，为使学生熟练掌握常用原始凭证的填写方法，需要进行模拟训练演示，而传统的手工模拟训练方式存在着一系列缺陷，难以适应会计教学及学习信息化发展方向的需要。为满足这一需要，运用 Excel2007 在 Excel VBA 平台下，开发完成原始凭证训练演示系统。

关键词：Excel VBA 编程；宏程序；会计单据

1 系统公共子程序编辑

打开 Visual Basic 编辑器，在模块 1 中编辑一系列公共子程序如下：

```
Public Sub 业务题选择(a As String)
Sheets(a).Unprotect
Dim i As Integer
i=Application.InputBox(prompt:= _
"请输入选做的业务题号:/(如 1、2 等,样题为 0)", Title:= _
"选择业务题", Type:=1)
If i > Range("B1").Value Then
MsgBox "所输入的题号超出已有业务题范围,请重新选择!"
Else
   Range("A4").Value = i
End If
Sheets(a).Protect
End Sub
Public Sub 样票显示(ws As String, unlockedrange _
As Range)
Application.ScreenUpdating = False
Sheets(ws).Unprotect
Dim a As Integer, yrange As Range, b As Integer
a = Range("iv3").End(xlToLeft).Column
b = Range("BB10").End(xlUp).Row
Set yrange = Range(Cells(3, 53), Cells(b, a))
yrange.Select
Selection.ClearContents
Dim r As Integer, c As Integer, c1 As Integer, _
myrange As Range
c = Range("iv22").End(xlToLeft).Column - 1
Set myrange = Range(Cells(22, 53), Cells(22, c))
myrange.Copy
Range("BA" & 3).PasteSpecial xlPasteAll
i = 23
c1 = 4
Do While Range("BB" & i) = Range("BB22")
Set myrange = Range(Cells(i, 53), Cells(i, c))
myrange.Copy
Range("BA" & c1).PasteSpecial xlPasteAll
i = i + 1
c1 = c1 + 1
Loop
  unlockedrange.Select
  Selection.Locked = True
  Sheets(ws).Protect
  Application.ScreenUpdating = True
  End Sub
Public Sub 票据填写(ws As String, _
unlockedrange As Range)
  Application.ScreenUpdating = False
  Sheets(ws).Unprotect
  unlockedrange.Select
  Selection.Locked = False
  unlockedrange.Value = ""
  Sheets(ws).Protect
  Application.ScreenUpdating = True
  End Sub
  Public Sub 票据删除(ws As String, _
  unlockedrange As Range)
  Application.ScreenUpdating = False
  Application.EnableEvents = False
  Sheets(ws).Unprotect
  If Range("BB13").Value = Range("BB22"). _
  Value Then
MsgBox "这是样票,不能删除!"
    GoTo bh
  End If
  Dim a As Integer, b As Integer, c As Integer, d _
  As Integer, e As Integer, f As Integer, aw As _
  Variant, myrange As Range, g As Integer
  a = Range("BB65536").End(xlUp).Row
  d = Range("IV22").End(xlToLeft).Column
  e = Range("AZ13").Value
```


DAICAR.COM
shop35833438.taobao.com

```
  For b = 23 To a
    If Range("BB" & b) = Range("BB13") Then
 aw = MsgBox (" 是否删除票据库中的该票据记录？ ", _
vbYesNo + vbQuestion, "删除提示")
    If aw = vbYes Then
        Exit For
    Else
        GoTo hh
    End If
End If
Next b
' 以下为删除票面记录库中的当前票据记录所在行
For f = 1 To e
 Set myrange = Range(Cells(b, 53), Cells(b, d))
 myrange.Delete shift:=xlShiftUp
Next f
' 以下为删除票面记录行后调整其后的顺序号
For c = b To a - e
  g = Cells(c, d).Value
  Cells(c, d) = g - 1
Next c
' 调整票面记录库区域的表格框线
Dim myborders As Borders, x As Long
Set myborders = Range("BA21").CurrentRegion.Borders
For x = xlEdgeLeft To xlInsideHorizontal
  With myborders(x)
      .LineStyle = xlContinuous
      .Weight = xlThin
      .ColorIndex = xlAutomatic
  End With
Next x
hh:
unlockedrange.Select
Selection.ClearContents
Selection.Locked = False
bh:
Sheets(ws).Protect
Application.EnableEvents = True
Application.ScreenUpdating = True
End Sub
Public Sub 显示上一张(ws As String, _
unlockedrange As Range)
Application.ScreenUpdating = False
Sheets(ws).Unprotect
Dim a As Integer, b As Integer, c As Integer, d _
As Integer, g As Integer
Dim e As Integer, myrang As Range, xrange _
As Range, f As Integer
a = Range("BB65536").End(xlUp).Row
b = Range("IV22").End(xlToLeft).Column
g = 0
Set xrange = Range(Cells(3, 53), Cells(10, b - 1))
xrange.ClearContents
For c = 22 To a
 If Range("BB" & c) = Range("BB13") Then
    g = g + 1
    d = Cells(c, b).Value - 1
    Exit For
 End If
  Next c
  If g = 0 Then
  Call 显示最后一张(ws, unlockedrange)
  Exit Sub
  End If
  If d <= 0 Then
  MsgBox "没有可显示票据! "
  GoTo hh
  Else
  f = 0
  For e = 23 To a
     If Cells(e, b) = d Then
        Set myrange = Range(Cells(e, 53), _
Cells(e, b - 1))
        myrange.Copy
        Range("BA" & 3 + f).PasteSpecial _ xlPasteValues
        f = f + 1
     End If
    Next e
    End If
    hh:
    unlockedrange.Select
    Selection.Locked = True
    Sheets(ws).Protect
    Application.ScreenUpdating = True
    End Sub
  Public Sub 显示下一张(ws As String, _
 unlockedrange As Range)
    Application.ScreenUpdating = False
    Sheets(ws).Unprotect
    Dim a As Integer, b As Integer, c As Integer, d _
    As Integer
 Dim e As Integer, myrang As Range, xrange As _
 Range, f As Integer
  a = Range("BB65536").End(xlUp).Row
  b = Range("IV22").End(xlToLeft).Column
  Set xrange = Range(Cells(3, 53), Cells(10, b - 1))
  xrange.ClearContents
  For c = 22 To a
 If Range("BB" & c) = Range("BB13") Then
    d = Cells(c, b).Value + 1
    Exit For
 End If
  Next c
  If d > Cells(a, b).Value Then
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
  MsgBox "没有可显示票据! "
  GoTo hh
   Else
  f = 0
  For e = 23 To a
     If Cells(e, b) = d Then
       Set myrange = Range(Cells(e, 53), _
Cells(e, b – 1))
         myrange.Copy
         Range("BA" & 3 + f).PasteSpecial _ xlPasteValues
         f = f + 1
      End If
   Next e
End If
hh:
unlockedrange.Select
Selection.Locked = True
Sheets(ws).Protect
Application.ScreenUpdating = True
End Sub
Public Sub 显示最后一张(ws As String, _
unlockedrange As Range)
Application.ScreenUpdating = False
Sheets(ws).Unprotect
Dim a As Integer, b As Integer, c As Integer, d As Integer
Dim e As Integer, myrang As Range, xrange As Range
a = Range("BB65536").End(xlUp).Row
b = Range("IV22").End(xlToLeft).Column
Set xrange = Range(Cells(3, 53), Cells(10, b – 1))
xrange.ClearContents
c = Cells(a, b).Value
e = 0
For d = 22 To a
    If Cells(d, b) = c Then
 Set myrange = Range(Cells(d, 53), Cells(d, b – 1))
  myrange.Copy
  Range("BA" & 3 + e).PasteSpecial xlPasteValues
  e = e + 1
   End If
Next d
unlockedrange.Select
Selection.Locked = True
Sheets(ws).Protect
Application.ScreenUpdating = True
End Sub
Public Sub 共性检验并保存(ws As String, _
unlockedrange As Range, nonullrange As Range)
Sheets(ws).Unprotect
Application.ScreenUpdating = False
Application.EnableEvents = False
Dim a As Range, b As String, c As Integer
Dim d As Integer,e As Integer, f As Integer
Dim g As Integer, h As Integer,i As Integer
Dim j As Integer, srange As Range
Dim wrange As Range, yrang As Range
Dim k As Integer, l As Integer, m As Integer
  For Each a In nonullrange
  If a.Value = "" Then
      MsgBox ("票据未填完整! 请继续填写! ")
      GoTo hh
  End If
  Next a
  b = InputBox _
("请输入业务题号(如果是任意填写,请输入"自由"字样! )", _
"题号确认")
  Range("BA13") = b
  If Range("BB13") = Range("BB22") Then
  MsgBox "这是样票号码,如要保存,请改换其他号码! "
  GoTo hh
  End If
  d = Range("BB65536").End(xlUp).Row
  e = Range("IV22").End(xlToLeft).Column
  g = 0
  For c = 23 To d
  If Cells(c, 54) = Cells(13, 54) Then
     j = c
     g = g + 1
  End If
  Next c
'以下 IF 程序将票面记录库中与当前票据号码相同的记录先删
'除再将当前票据记录写入库中同样位置;或将新的票据记录写
'入票面记录库最后
  If g > 0 Then
  Set srange = Range(Cells(j – g + 1, 53), _
Cells(j, e))
  srange.Delete shift:=xlUp
  i = Range("AZ13").Value
  For h = 1 To i
    Set wrange = Range(Cells(j – g + 1, 53), _
Cells(j – g + 1, e))
    wrange.Insert shift:=xlDown
  Next h
  Set yrange = Range(Cells(13, 53), _
  Cells(13 + i – 1, e – 1))
  yrange.Copy
  Range("BA" & j – g + 1).PasteSpecial xlPasteValues
  Else
  i = Range("AZ13").Value
  Set yrange = Range(Cells(13, 53), _
  Cells(13 + i – 1, e – 1))
  yrange.Copy
  Range("BA" & d + 1).PasteSpecial xlPasteValues
  End If
'调整票面记录库的记录顺序号
d = Range("BB65536").End(xlUp).Row
```


LAICAR.COM
shop35833438.taobao.com

```
For k = 23 To d
  If Cells(k, e) = "" Then
     l = Cells(k – 1, e).Value + 1
     Exit For
  End If
Next k
For k = 23 To d
  If Cells(k, e) = "" Then
     Cells(k, e) = l
  End If
Next k
Dim r As Borders, x As Long
    Set r = Range("BA21").CurrentRegion.Borders
     For x = xlEdgeLeft To xlInsideHorizontal
        With r(x)
        .LineStyle = xlContinuous
        .Weight = xlThin
       .ColorIndex = xlAutomatic
           End With
       Next x
hh:
unlockedrange.Select
Selection.Locked = True
  Application.EnableEvents = True
  Sheets(ws).Protect
  Application.ScreenUpdating = True
End Sub
Public Sub 票据修改(ws As String, _
unlockedrange As Range)
Application.ScreenUpdating = False
Sheets(ws).Unprotect
If Range("BB13") = Range("BB22") Then
 MsgBox "样票不能修改！"
 Sheets(ws).Protect
 Exit Sub
End If
unlockedrange.Select
Selection.Locked = False
Sheets(ws).Protect
Application.ScreenUpdating = True
End Sub
  Public Sub 退回主界面(ws As String)
  Sheets("主界面").Visible = True
  Sheets(ws).Visible = False
  Application.DisplayFullScreen = True
  ActiveWindow.DisplayGridlines = False
  End Sub
```

2 窗体控件指定宏程序编辑

激活“支票”工作表，指定“显示样票”控件的宏程序：

```
Sub 按钮 10_Click()
Call 样票显示("支票", Range("支票未锁定区域"))
End Sub
```

指定“填写支票”控件的宏程序：

```
Sub 按钮 11_Click()
Call 票据填写("支票", Range("支票未锁定区域"))
End Sub
```

指定“删除”控件的宏程序：

```
Sub 按钮 12_Click()
Call 票据删除("支票", Range("支票未锁定区域"))
End Sub
```

指定“显示上一张”控件的宏程序：

```
Sub 按钮13_Click()
Call 显示上一张("支票", Range("支票未锁定区域"))
End Sub
```

指定“显示下一张”控件的宏程序：

```
Sub 按钮14_Click()
Call 显示下一张("支票", Range("支票未锁定区域"))
End Sub
```

指定“最末一张”控件的宏程序：

```
Sub 按钮20_Click()
Call 显示最后一张("支票", Range("支票未锁定区域"))
End Sub
```

指定“修改”控件的宏程序：

```
Sub 按钮 19_Click()
Call 票据修改("支票", Range("支票未锁定区域"))
End Sub
```

指定“退回主界面”控件的宏程序：

```
Sub 按钮16_Click()
Call 退回主界面("支票")
End Sub
```

指定“业务题选择”控件的宏程序：

```
Sub 按钮 9_Click()
  Call 业务题选择("支票")
End Sub
```

“增值税发票”工作表中的这些控件对应程序编辑，只需将上述程序中的“支票”字样改为“增值税发票”字样，其他原始凭证工作表也是将“支票”字样改为相应原始凭证工作表的名称。各原始凭证工作表中“保存”控件对应程序包含个性检验部分、共性检验部分及保存部分，其中的个性检验部分需针对各原始凭证的特定填写要求分别设计。

支票“保存”控件指定宏程序如下：

```
Sub 按钮 15_Click()
'以下为支票个性检验部分
Sheets("支票").Unprotect
If Range("L33") <> Range("M12") Or _
Range("M33") <> Range("Q12") Or _
Range("M34") <> Range("T12") Then
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
    MsgBox ("日期输入存根与票面不符！")
    Sheets("支票").Protect
    Exit Sub
End If
If Range("C11") <> Range("AA11") Then
    MsgBox ("支票号码存根与票面不符！")
    Sheets("支票").Protect
    Exit Sub
End If
If Range("C21") <> Range("M13") Then
    MsgBox ("收款人存根与票面不符！")
    Sheets("支票").Protect
    Exit Sub
End If
If Range("C22") <> Range("C28") Then
    MsgBox ("支票金额存根与票面不符！")
    Sheets("支票").Protect
    Exit Sub
End If
If Range("C23") <> Range("M18") Then
   MsgBox ("支票用途存根与票面不符！")
   Sheets("支票").Protect
   Exit Sub
End If
If Range("L28") <> Range("M14") Then
   MsgBox ("支票金额大写有错与数值金额不符！")
   Sheets("支票").Protect
   Exit Sub
End If
   '以下是共性检验部分及保存部分
Call 共性检验并保存("支票", _
Range("支票未锁定区域"), Range("zp 文字不空区域"))
End Sub
```

增值税发票工作表的“保存”控件对应宏程序如下：

```
Sub 按钮8_Click()
Sheets("增值税发票").Unprotect
Dim r As Integer, n As Integer
r = Range("AZ13").Value
For n = 1 To r
    If Range("BM" & 12 + n) <> _
Range("BK" & 12 + n) * Range("BL" & 12 + n) Then
  MsgBox ("第" & n & _
    "行货物的金额不等于数量与单价乘积！")
    Sheets("增值税发票").Protect
    Exit Sub
  End If
Next n
For n = 1 To r
      If Range("BO" & 12 + n) <> Range("BM" & 12 + n) * _
        Range("BN" & 12 + n) / 100 Then
     MsgBox ("第" & n & _
        "行货物的税额不等于金额与税率乘积！")
      Sheets("增值税发票").Protect
      Exit Sub
     End If
Next n
If Range("BP13") <> "" Then
  If Range("BP13") <> WorksheetFunction. _
     sum(Range("BK13:BK16")) Then
     MsgBox ("数量合计不符！")
     Sheets("增值税发票").Protect
     Exit Sub
  End If
End If
If Range("BQ13") <> WorksheetFunction. _
sum(Range("BM13:BM16")) Then
   MsgBox ("金额合计不等！")
   Sheets("增值税发票").Protect
   Exit Sub
End If
If Range("BR13") <> WorksheetFunction. _
sum(Range("BO13:BO16")) Then
   MsgBox ("税额合计不等！")
   Sheets("增值税发票").Protect
   Exit Sub
End If
If Range("BT13") <> WorksheetFunction. _
sum(Range("BR13:BQ13")) Then
   MsgBox ("价税合计不等！")
   Sheets("增值税发票").Protect
   Exit Sub
End If
If Range("E21") <> Range("E30") Then
   MsgBox ("中文大写与价税金额不符！")
   Sheets("增值税发票").Protect
   Exit Sub
End If
Call 共性检验并保存("增值税发票", _
Range("增值税发票未锁定区域"), Range _
("zzsfp 文字不空区域"))
End Sub
```

（收稿日期：2013-01-08）

LAICAR.COM
shop35833438.taobao.com

值班室信息查询系统设计与实现

刘 烽

摘 要：针对目前企事业单位值班室信息查询工作日趋繁重的现实，设计并实现了一款功能强大、使用简单的值班室信息查询系统。系统实现了对人员、车辆、电话号码等热点信息的管理与查询，较好地提高了值班人员工作效率。

关键词：值班室；信息查询；数据库

1 引言

值班室做为企事业单位的重要部位，具有信息的上传下达、人员和车辆管理、安防监控、突发情况处置等职能。目前，随着值班室职能的不断拓展，值班人员面临的日常值班工作也日趋繁重、复杂。同时，值班人员需要实时掌握本单位相关信息的动态变化情况，以便及时准确地向上级领导反馈。因此，值班员如何快速、高效地从各种纷繁的信息中查找出上级领导关注的热点信息就显得尤为重要，而开发一款功能强大、操作简单的值班室信息查询系统对于提高值班员的工作效率，提高对各类信息的掌控能力，具有重要作用。

本系统以 Visual Basic 为集成开发环境，Access 作为数据库引擎，主要实现对人员、电话号码、车辆出行等重点监控目标的信息查询功能。将该系统应用到实际的日常值班值勤之后，值班值勤人员的工作效率得到了明显的改善。

2 系统结构

系统结构如图 1 所示。

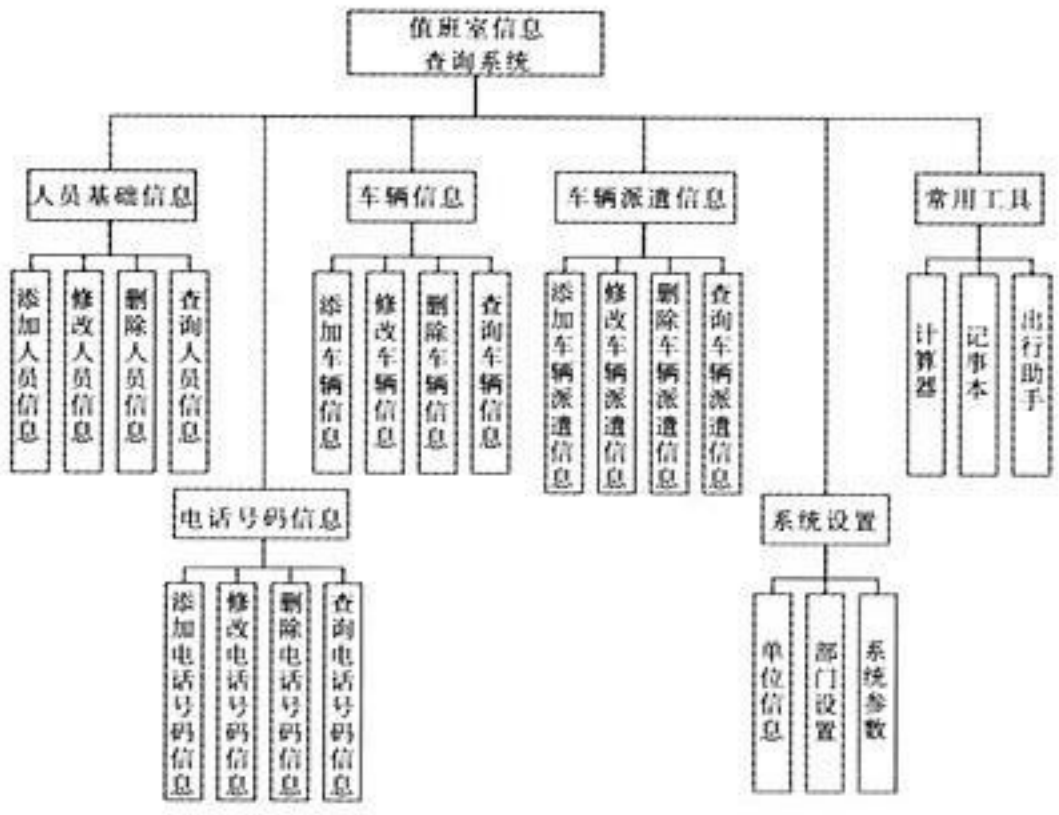


图 1 值班室信息查询系统结构图

3 系统设计与实现

系统运行后界面如图 2 所示，下面介绍各分子系统的具体实现。



图 2 值班室信息查询系统运行界面

3.1 系统登录界面 (frmLogin.frm)

登录主界面如图 3 所示。



图 3 系统登陆界面

```
Private Const WS_EX_LAYERED = &H80000
Private Const GWL_EXSTYLE = (-20)
Private Const LWA_ALPHA = &H2
Private Const LWA_COLORKEY = &H1
Dim Alpha As Integer ' 透明度
 Private Declare Function GetWindowLong Lib "user32"
Alias "GetWindowLongA" (ByVal hwnd As Long, ByVal
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
nIndex As Long) As Long
    Private Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA" (ByVal hwnd As Long, ByVal nIndex As Long, ByVal dwNewLong As Long) As Long
    Private Declare Function SetLayeredWindowAttributes Lib "user32" (ByVal hwnd As Long, ByVal crKey As Long, ByVal bAlpha As Byte, ByVal dwFlags As Long) As Long
    Private Sub Form_Initialize()
    Call InitCommonControls
End Sub
Private Sub Form_Load()
Me.Shape1.Top = Me.Top
Me.Shape1.Left = Me.Left
Me.Shape1.Width = Me.ScaleWidth
Me.Shape1.Height = Me.ScaleHeight
'-----------------------------------
Dim Ret As Long
   Ret = GetWindowLong(Me.hwnd, GWL_EXSTYLE)
   Ret = Ret Or WS_EX_LAYERED
   SetWindowLong Me.hwnd, GWL_EXSTYLE, Ret
   Timer1.Interval = 20
'-----------------------------------
Call OpenConn
SQL = "select czy from 系统用户表"
rs.Open SQL, cn, 1, 1
   Do While Not rs.EOF
      Me.Combo1.AddItem rs! czy
      rs.MoveNext
   Loop
Call CloseConn
Call OpenConn
SQL = "select 单位名称 from 单位信息"
rs.Open SQL, cn, 1, 1
   If rs.RecordCount > 0 Then
      gDepartName = rs! 单位名称
   Else
      gDepartName = "单位信息不可知"
   End If
Call CloseConn
End Sub
Private Sub Image2_Click()
'Me.cmdDL = True
    If Me.Combo1.Text = "" Then
     MsgBox "请选择登陆用户！", 48, "错误提示"
     Exit Sub
    End If
    Call OpenConn
    SQL = "select * from 系统用户表 where Czy='" & Me.Combo1.Text & "'"
    rs.Open SQL, cn, 1, 1
     If CovertStr (txtPass.Text) = IIf (Trim (IsNull (rs.Fields("pass"))), "", Trim(rs.Fields("pass"))) Then
      gAdmin = rs.Fields("czy")
      gAdminPowers = rs.Fields("czyqx")
      Call CloseConn
      MDIfrm.Show
      Unload Me
    Else
      MsgBox "密码错误！", 48, "错误提示"
    End If
End Sub
    Private Sub Image3_Click()
    End
    End Sub
    Private Sub Timer1_Timer()
    Alpha = Alpha + 5
    If Alpha > 255 Then
      Timer1.Enabled = False
    Exit Sub
    End If
        SetLayeredWindowAttributes Me.hwnd, 0, Alpha, LWA_ALPHA
    End Sub
```

3.2 人员基础信息查询 (frmPersonBaseData.frm)

人员基础信息运行界面如图 4 所示。

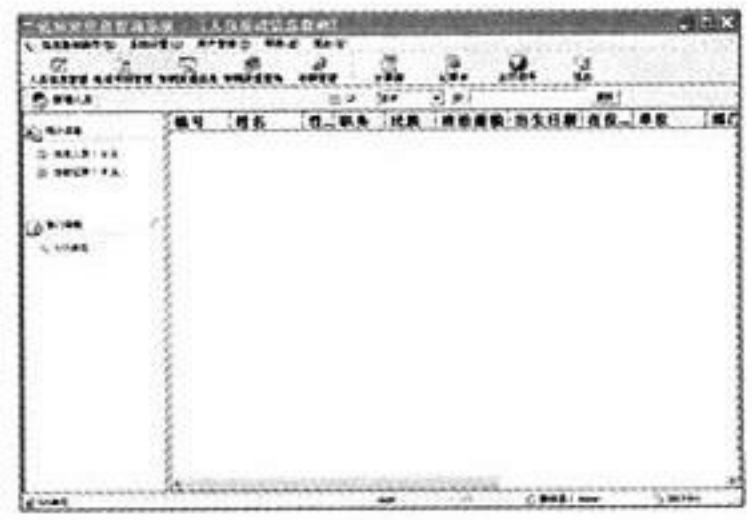


图 4 人员基础信息查询

```
Dim SQL As String
Dim addLVW As ListItem
Private Sub Check1_Click()
Me.ListView1.Gridlines = Me.Check1.Value
End Sub
Private Sub Check2_Click()
If Me.Check2.Value = 1 Then
   For i = 1 To Me.TreeView1.Nodes.Count
      Me.TreeView1.Nodes(i).Expanded = True
   Next
Else
   For i = 1 To Me.TreeView1.Nodes.Count
      Me.TreeView1.Nodes(i).Expanded = False
   Next
End If
End Sub
Private Sub cmdSeek_Click()
```


LAICAR.COM shop35833438.taobao.com

```
If Trim(Me.txtSeek.Text) = "" Then
    MsgBox "请输入要查找的职工姓名！", 48, "提示"
    Me.txtSeek.SetFocus
    Exit Sub
End If
SQL = "select zgid,编号,姓名,性别,职务,民族,政治面貌,出生日
期,在位情况,单位,部门,联系电话,移动电话,最高学历,职称,来
源 from 基础档案表"
SQL = SQL & " where 姓名 like '%" & Me.txtSeek.Text &
"%' ORDER BY 编号 ASC"
Call UpdateList
End Sub
Private Sub cmdHidenCommad_Click()
Call UpdateList '用于刷新本窗口 listview,刷新时使用了本窗
'口的变量 SQL
End Sub
Private Sub Combo1_Click()
If Me.Combo1.Text = "全部" Then
    SQL = "select zgid,编号,姓名,性别,职务,民族,政治面貌,出
生日期,在位情况,单位,部门,联系电话,移动电话,最高学历,职
称,来源 from 基础档案表"
    SQL = SQL & " ORDER BY 编号 ASC"
Else
    SQL = "select zgid,编号,姓名,性别,职务,民族,政治面貌,出
生日期,在位情况,单位,部门,联系电话,移动电话,最高学历,职
称,来源 from 基础档案表"
    SQL = SQL & " where 在位情况='" & Me.Combo1.Text &
"' ORDER BY 编号 ASC"
End If
Call UpdateList
End Sub
Private Sub Form_Initialize()
Call InitCommonControls 'XP 效果
End Sub
Private Sub Form_Load()
Me.Caption = "人员基础信息查询"
'-------------------------
'统计信息
Call OpenConn
SQL = "select count(*) as zgzs from 基础档案表"
rs.Open SQL, cn, 1, 1
    Me.labCONUT.Caption = rs! zgzs & " 人"
Call CloseConn
'更新树视图
Call UpdateTreeView(Me.TreeView1)
'livsview 设置
With Me.ListView1
    .ColumnHeaders.Add = "编号"
    .ColumnHeaders.Add = "zgid"
    .ColumnHeaders.Add = "姓名"
    .ColumnHeaders.Add = "性别"
    .ColumnHeaders.Add = "职务"
    .ColumnHeaders.Add = "民族"
    .ColumnHeaders.Add = "政治面貌"
    .ColumnHeaders.Add = "出生日期"
    .ColumnHeaders.Add = "在位情况"
    .ColumnHeaders.Add = "单位"
    .ColumnHeaders.Add = "部门"
    .ColumnHeaders.Add = "联系电话"
    .ColumnHeaders.Add = "移动电话"
    .ColumnHeaders.Add = "最高学历"
    .ColumnHeaders.Add = "职称"
    .ColumnHeaders.Add = "来源"
    .ColumnHeaders(1).Width = 1200
    .ColumnHeaders(2).Width = 0
    .ColumnHeaders(4).Width = 600
    .ColumnHeaders(5).Width = 1000
    .ColumnHeaders(6).Width = 1000
    .ColumnHeaders(9).Width = 1000
End With
Call OpenConn
SQL = "select * from 参数设置表 where 项目='在位情况'"
rs.Open SQL, cn, 1, 1
Do While Not rs.EOF
    Me.Combo1.AddItem rs! 值
    rs.MoveNext
Loop
Call CloseConn
Me.Combo1.AddItem "全部"
Me.Combo1.ListIndex = 0
End Sub
Private Sub Form_Resize() '窗体内控件尺寸
If Me.ScaleHeight > 8295 Or Me.ScaleWidth > 11280 Then
    Me.ListView1.Height = Me.ScaleHeight - Me.ListView1.
Top
    Me.ListView1.Width = Me.ScaleWidth - Me.ListView1.
Left
   Me.PicLEFT.Height = Me.ScaleHeight - Me.PicLEFT.Top
   Me.TreeView1.Height = Me.PicLEFT.Height - Me.Shape1
(1).Top - 350
End If
Me.Picture1.Left = Me.ScaleWidth - Me.Picture1.Width -
240
End Sub
Private Sub ListView1_DblClick()
    If Me.ListView1.ListItems.Count <= 0 Then Exit Sub
    gLstSelID = Me.ListView1.SelectedItem.SubItems(1)
    newFlag = 1
    frmPersonalInfo.Show 1
End Sub
Private Sub Toolbar1_ButtonClick (ByVal Button As
MSComctlLib.Button)
Select Case Button.Index
    Case Is = 2 '新增职工
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
        newFlag = 0
        frmPersonalInfo.Show 1
        Me.picLOAD.Left = (Me.ScaleWidth + Me.PicLEFT.
Width – Me.picLOAD.Width) / 2
        Me.picLOAD.Top = Me.ScaleHeight / 2 – Me.
picLOAD.Height
        Me.picLOAD.Visible = True
        DoEvents ' 释放控制权
        Dim i As Integer
        Dim j As Integer
        i = 3
        Call OpenConn
            If Me.Combo1.Text = "全部" Then
    SQL1 = "select * from 基础档案表 ORDER BY 编号 ASC"
            Else
            SQL1 = "select * from 基础档案表 where 在位情
况='" & Me.Combo1.Text & "' ORDER BY 编号 ASC"
            End If
            rs.Open SQL1, cn, 1, 1
            Do While Not rs.EOF
                For j = 1 To 33
                    xlsheet.Cells(i, j) = rs.Fields(j).Value
                Next
                rs.MoveNext
                i = i + 1
            Loop
            Call CloseConn
            Me.picLOAD.Visible = False
End Select
End Sub
Private Sub UpdateList()
Me.picLOAD.Left = (Me.ScaleWidth + Me.PicLEFT.Width –
Me.picLOAD.Width) / 2
Me.picLOAD.Top = Me.ScaleHeight / 2 – Me.picLOAD.
Height
Me.picLOAD.Visible = True
DoEvents ' 释放控制权
Me.ListView1.ListItems.Clear
Call OpenConn
    rs.Open SQL, cn, 1, 1
    Do While Not rs.EOF
        If rs! 性别 = "男" Then tb = 1
        If rs! 性别 = "女" Then tb = 2
        Set addLVW = Me.ListView1.ListItems.Add (, , rs! 编
号, , tb)
            addLVW.SubItems(1) = rs! ZGID
            addLVW.SubItems(2) = rs! 姓名
            addLVW.SubItems(3) = rs! 性别
            addLVW.SubItems(4) = rs! 职务
            addLVW.SubItems(5) = rs! 民族
            addLVW.SubItems(6) = rs! 政治面貌
            addLVW.SubItems(7) = rs! 出生日期
            addLVW.SubItems(8) = rs! 在位情况
            addLVW.SubItems(9) = rs! 单位
            addLVW.SubItems(10) = rs! 部门
            addLVW.SubItems(11) = rs! 联系电话
            addLVW.SubItems(12) = rs! 移动电话
            addLVW.SubItems(13) = rs! 最高学历
            addLVW.SubItems(14) = rs! 职称
            addLVW.SubItems(15) = rs! 来源
            rs.MoveNext
        Loop
        Me.labDQJL.Caption = rs.RecordCount & " 人"
        Call CloseConn

Me.picLOAD.Visible = False
End Sub
Private Sub TreeView1_NodeClick (ByVal Node As
MSComctlLib.Node)
Dim strPersonRemain As String ' 在位情况
Dim treeFlag As String
treeFlag = Left(Node.Key, 1)
If Me.Combo1.Text = "全部" Then
    strPersonRemain = ""
Else
    strPersonRemain = Me.Combo1.Text
End If
Select Case treeFlag
    Case Is = "A"
        SQL = "select zgid,编号,姓名,性别,职务,民族,政治面
貌,出生日期,在位情况,单位,部门,联系电话,移动电话,最高学历,
职称,来源 from 基础档案表"
        SQL = SQL & " where 在位情况 like '%" &
strPersonRemain & "%' ORDER BY 编号 ASC"
        Call UpdateList
    Case Is = "B"
        SQL = "select zgid,编号,姓名,性别,职务,民族,政治面
貌,出生日期,在位情况,单位,部门,联系电话,移动电话,最高学历,
职称,来源 from 基础档案表"
        SQL = SQL & " where 单位='" & Node.Text & "' and
在位情况 like '%" & strPersonRemain & "%' ORDER BY 编
号 ASC"
        Call UpdateList
    Case Is = "C"
        treepath = TreeView1.SelectedItem.FullPath 'tree 完
'整路径
        a = InStr(1, treepath, "\") ' 找第一个 \ 分隔位置
        b = InStr(a + 1, treepath, "\") ' 找第二个 \ 分隔位置
        c = Mid(treepath, a + 1, b – a – 1) ' 将单位取出
        SQL = "select zgid,编号,姓名,性别,职务,民族,政治面
貌,出生日期,在位情况,单位,部门,联系电话,移动电话,最高学历,
职称,来源 from 基础档案表"
        SQL = SQL & " where 单位='" & c & "' and 部门='"
& Node.Text & "' and 在位情况 like '%" & strPersonRemain
```


LAICAR.COM
shop35833438.taobao.com

```
& "%' ORDER BY 编号 ASC"
        Call UpdateList
End Select
End Sub
```

3.3 电话号码信息查询 (frmNumberInfo.frm)

电话号码信息查询运行界面如图 5 所示。



图 5 电话号码信息查询

```
Dim SQL As String
Dim addLVW As ListItem
Dim Dqyd As String
Private Sub Check1_Click()
Me.ListView1.Gridlines = Me.Check1.Value
End Sub
Private Sub Check2_Click()
If Me.Check2.Value = 1 Then
   For i = 1 To Me.TreeView1.Nodes.Count
      Me.TreeView1.Nodes(i).Expanded = True
   Next
Else
   For i = 1 To Me.TreeView1.Nodes.Count
      Me.TreeView1.Nodes(i).Expanded = False
   Next
End If
End Sub
Private Sub cmdSeek_Click()
If Option1.Value = True Then
If Trim(Me.txtSeek.Text) = "" Then
   MsgBox "请输入要查找的人员姓名！", 48, "提示"
   Me.txtSeek.SetFocus
   Exit Sub
End If
SQL = "select * from PhoneList where 姓名 like '%" & Me.
txtSeek.Text & "%'"
Call UpdateList
Else
 If Trim(Me.txtSeek.Text) = "" Then
   MsgBox "请输入要查找的电话号码！", 48, "提示"
   Me.txtSeek.SetFocus
   Exit Sub
End If
SQL = "select * from PhoneList where 办公室号码='" & Me.
txtSeek.Text & "'"
SQL = SQL + "or 宿舍号码='" + Me.txtSeek.Text + "'"
SQL = SQL + "or 手机='" + Me.txtSeek.Text + "'"
SQL = SQL + "or 虚拟号码='" + Me.txtSeek.Text + "'"
SQL = SQL + "or 外拨号码='" + Me.txtSeek.Text + "'"
Call UpdateList
End If
End Sub
Private Sub cmdExtraCommand_Click()
Call UpdateList '用于刷新本窗体
End Sub
Private Sub Combo1_Click()
'SQL = "select * from PhoneList where 月度='" & Me.
Combo1.Text & "'"
SQL = "select * from PhoneList " & " ORDER BY ID ASC"
Call UpdateList
End Sub
Private Sub Form_Load()
Me.Caption = "电话号码查询"
If IsNameTag Then
   Option1.Value = True
   Option2.Value = False
Else
   Option1.Value = False
   Option2.Value = True
End If
SQL = "select ID,姓名,办公室号码,宿舍号码,手机,虚拟号码,外
拨号码 from PhoneList "
 SQL = SQL & "ORDER BY ID ASC"
'设置 lvw----------------------------
With Me.ListView1
.ColumnHeaders.Add = "姓名"
.ColumnHeaders.Add = "序号"
.ColumnHeaders.Add = "办公室号码"
.ColumnHeaders.Add = "宿舍号码"
.ColumnHeaders.Add = "手机"
.ColumnHeaders.Add = "虚拟号码"
.ColumnHeaders.Add = "外拨号码"
.ColumnHeaders(1).Width = 3200
.ColumnHeaders(2).Width = 0
.ColumnHeaders(3).Width = 3200
.ColumnHeaders(4).Width = 3200
.ColumnHeaders(5).Width = 3200
.ColumnHeaders(6).Width = 3200
.ColumnHeaders(7).Width = 3200
End With
Call UpdateList
End Sub
Private Sub Form_Initialize()
Call InitCommonControls 'XP 效果
End Sub
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
Private Sub Form_Resize() ′窗体内控件尺寸
If Me.ScaleHeight > 8295 Or Me.ScaleWidth > 11280 Then
   Me.ListView1.Height = Me.ScaleHeight - Me.ListView1.
Top
   Me.ListView1.Width = Me.ScaleWidth - Me.ListView1.
Left
  Me.PicLEFT.Height = Me.ScaleHeight - Me.PicLEFT.Top
End If
Me.Picture1.Left = Me.ScaleWidth - Me.Picture1.Width -
240
End Sub
Private Sub ListView1_DblClick()
If Me.ListView1.ListItems.Count <= 0 Then Exit Sub
gLstNumerID = CInt (Me.ListView1.SelectedItem.SubItems
(1))
frmNumberManage.Show 1
End Sub
Private Sub Option1_Click()
IsNameTag = True
End Sub
Private Sub Option2_Click()
IsNameTag = False
End Sub
Private Sub Toolbar1_ButtonClick (ByVal Button As
MSComctlLib.Button)
Select Case Button.Index
   Case Is = 2
      gLstNumerID = 0
      gIsNew = 0
      frmNumberManage.Show 1
End Select
End Sub
Private Sub UpdateList()
Me.picLOAD.Left = (Me.ScaleWidth + Me.PicLEFT.Width -
Me.picLOAD.Width) / 2
Me.picLOAD.Top = Me.ScaleHeight / 2 - Me.picLOAD.
Height
Me.picLOAD.Visible = True
DoEvents ′释放控制权
Me.ListView1.ListItems.Clear
Call OpenConn
rs.Open SQL, cn, 1, 1
Do While Not rs.EOF
  Set addLVW = Me.ListView1.ListItems.Add(, , rs! 姓名, , 2)
      addLVW.SubItems(1) = (rs! ID) & ""
      addLVW.SubItems(2) = (rs! 办公室号码) & ""
      addLVW.SubItems(3) = (rs! 宿舍号码) & ""
      addLVW.SubItems(4) = (rs! 手机) & ""
      addLVW.SubItems(5) = (rs! 虚拟号码) & ""
      addLVW.SubItems(6) = (rs! 外拨号码) & ""
      rs.MoveNext
Loop
Me.labCONUT.Caption = rs.RecordCount
Call CloseConn
Me.picLOAD.Visible = False
End Sub
```

3.4 车辆信息查询 (frmCarInfoManage.frm)

车辆信息查询运行界面如图 6 所示。



图 6 车辆信息查询

```
Dim i As Integer
Dim c          ′判断 MsgBox 函数执行哪个按钮的变量
Public StrCodes As PeopleCode ′身份证号码
Private Sub cbxCar_KeyDown (KeyCode As Integer, Shift As
Integer)
  If KeyCode = 13 Then
     Text1(2).SetFocus
  End If
End Sub
Private Sub dgrContent_Click()
  On Error Resume Next
  If Adodc1.Recordset.RecordCount > 0 Then
      Text1(0).Text = Adodc1.Recordset.Fields("carid")
      Text1(1).Text = Adodc1.Recordset.Fields("carnumber")
      cbxCar.Text = Adodc1.Recordset.Fields("carclass")
      Text1(2).Text = Adodc1.Recordset.Fields("drivername")
      DT2.Value = Adodc1.Recordset.Fields("birthdate")
      Text1(3).Text = Adodc1.Recordset.Fields("officer")
      Text1(4).Text = Adodc1.Recordset.Fields("drivercard")
      Text1(5).Text = Adodc1.Recordset.Fields("email")
        Text1 (6).Text = Adodc1.Recordset.Fields ("
officernumber")
      Text1(7).Text = Adodc1.Recordset.Fields("officerphone")
      Text1(8).Text = Adodc1.Recordset.Fields("driverphone")
      Text1(9).Text = Adodc1.Recordset.Fields("postcode")
      Text1(10).Text = Adodc1.Recordset.Fields("address")
      Text1(11).Text = Adodc1.Recordset.Fields("appendix")
  End If
  For i = 0 To 11
    Text1(i).Locked = False
  Next i
End Sub
```


LAICAR.COM
shop35833438.taobao.com

```
Private Sub DT2_KeyDown (KeyCode As Integer, Shift As Integer)
  If KeyCode = 13 Then
    Text1(4).SetFocus
  End If
End Sub
Private Sub Form_Load()
   Me.Left = (Screen.Width – Me.Width) / 2
   Me.Top = (Screen.Height – Me.Height) / 2
   Call LoadFile
   For i = 1 To 11      '锁定文本框,禁止输入字符信息
     Text1(i).Locked = True
   Next i
     Adodc1.ConnectionString = "Provider =Microsoft.Jet.OLEDB.4.0;Data Source=" & App.Path & "\data\data.mdb;Jet OLEDB:Database Password =123456;Persist Security Info = False;"
   Adodc1.RecordSource = "select * from tbCarInfo order by carid"
   Adodc1.Refresh
   DT2.Value = Date    '使日期控件显示当前系统的日期时间
       Call DBGCon
   With tlbCarInfo
        .Buttons(1).Enabled = True
        .Buttons(2).Enabled = True
        .Buttons(3).Enabled = True
        .Buttons(4).Enabled = False
        .Buttons(5).Enabled = True
        .Buttons(6).Enabled = True
     End With
     Me.cbxCar.AddItem "越野车"
     Me.cbxCar.AddItem "大客车"
     Me.cbxCar.AddItem "工程机械车"
     Me.cbxCar.AddItem "其它"
End Sub
Private Sub tlbCarInfo_ButtonClick (ByVal Button As MSComctlLib.Button)
  Select Case Button.Index
    Case 1
   With tlbCarInfo
        .Buttons(1).Enabled = False
        .Buttons(2).Enabled = True
        .Buttons(3).Enabled = True
        .Buttons(4).Enabled = True
        .Buttons(5).Enabled = True
        .Buttons(6).Enabled = True
End With
     For i = 1 To 11
       Text1(i).Locked = False
       Text1(i).Text = ""
       Text1(1).SetFocus
     Next i
        AdoRs.Open "select * from tbCarInfo order by carid", Cnn, adOpenKeyset
          If AdoRs.RecordCount > 0 Then
            AdoRs.MoveLast
              StrNum = Mid (AdoRs.Fields ("carid"), 2, Len (AdoRs.Fields("carid")))
              Call CovertNum      '调用位数转换函数
              Text1(0).Text = "C" & StrIn
           Else
              Text1(0).Text = "C0001"
          End If
       AdoRs.Close
    Case 2                        '删除信息
       Call Deletes
       Call DBGCon
    Case 3                        '修改信息
       Call Edits
       Call DBGCon
    Case 4                        '保存信息
      Call Saves
      Call DBGCon
    Case 5
        With tlbCarInfo
       .Buttons(1).Enabled = True
       .Buttons(2).Enabled = True
       .Buttons(3).Enabled = True
       .Buttons(4).Enabled = False
       .Buttons(5).Enabled = True
       .Buttons(6).Enabled = True
     End With
       For i = 1 To 11
         Text1(i).Text = ""
         Text1(1).SetFocus
       Next i
       Adodc1.RecordSource = "select * from tbCarInfo order by carid"
       Adodc1.Refresh
       Call DBGCon
     Case 6
       frmQueryItem.Show 1
     Case 7
       Unload Me
  End Select
End Sub
Private Sub Text1_KeyDown (Index As Integer, KeyCode As Integer, Shift As Integer)
  On Error Resume Next
    If Index = 0 Then
      If KeyCode = 13 Then
        Adodc1.RecordSource = "select * from tbCarInfo where carid='" + Text1(0).Text + "'"
        Adodc1.Refresh
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
        If Adodc1.Recordset.RecordCount > 0 Then
            Text1(0).Text = Adodc1.Recordset.Fields("carid")
    Text1(1).Text = Adodc1.Recordset.Fields("carnumber")
            cbxCar.Text = Adodc1.Recordset.Fields("carclass")
                    Text1  (2).Text = Adodc1.Recordset.Fields("
drivername")
            DT2.Value = Adodc1.Recordset.Fields("birthdate")
            Text1(3).Text = Adodc1.Recordset.Fields("officer")
                    Text1(4).Text = Adodc1.Recordset.Fields("
drivercard")
            Text1(5).Text = Adodc1.Recordset.Fields("email")
                    Text1(6).Text = Adodc1.Recordset.Fields("
officernumber")
                    Text1(7).Text = Adodc1.Recordset.Fields("
officerphone")
                    Text1  (8).Text = Adodc1.Recordset.Fields("
driverphone")
                    Text1(9).Text = Adodc1.Recordset.Fields("
postcode")
                    Text1(10).Text = Adodc1.Recordset.Fields("
address")
                    Text1  (11).Text = Adodc1.Recordset.Fields("
appendix")
         End If
       For i = 1 To 11
        Text1(i).Locked = False
      Next i
      End If
End If
 Call DBGCon
 If KeyCode = 13 Or KeyCode = 40 Then
If Index = 1 Then
    cbxCar.SetFocus
    Exit Sub
End If
If Index = 2 Then
  If KeyCode = 13 Then
  End If
     Exit Sub
End If
If Index = 4 Then      '限定身份证号
   Set StrCodes = New PeopleCode
    StrCodes.PeopleCode (Text1(4).Text)
    DT2.Value = StrCodes.PeopleBirth
    Text1(4).Text = StrCodes.PeopleCodes
  End If
  If Index = 3 Then
     DT2.SetFocus
     Exit Sub
  End If
   If Index >= 0 Then Text1(Index + 1).SetFocus
End If
If Index = 4 Then
  If KeyCode = 38 Then
    DT2.SetFocus
    Exit Sub
  End If
End If
If Index = 2 Then
  If KeyCode = 38 Then
    cbxCar.SetFocus
     Exit Sub
   End If
  End If
  If Index = 11 Then
   If KeyCode = 13 Then
     Call Saves
   End If
  End If
  If Index >= 0 And KeyCode = 40 Then Text1  (Index + 1).
SetFocus
  If Index <= 11 And KeyCode = 38 Then Text1  (Index – 1).
SetFocus
End Sub
Private Sub Saves()    '保存信息的事件过程
 If Text1(1).Text = "" Or Text1(3).Text = "" Then
    MsgBox "重要信息不能为空值", 48, "提示信息"
 Else
      AdoRs.Open  "select  *  from  tbCarInfo  where
carnumber='" + Text1(1).Text + "'", Cnn, adOpenKeyset
      If AdoRs.RecordCount > 0 Then                '判断要
保存的信息是否已经存在
         MsgBox "该车辆信息已经存在", 48, "提示信息"
         AdoRs.Close
      Else
         AdoRs.Close
        c = MsgBox("确认保存信息吗", 33, "提示信息")
         If c = vbOK Then                          '如果选择的
是"确定"按钮则保存信息
           Set AdoRs = Cnn.Execute  ("insert into tbCarInfo
values('" & Text1(0) & "','" & Text1(1) & "','" & cbxCar & "','" &
Text1(2) & "','" & Text1(3) & "','" _
           & Text1(6) & "','" & Text1(4) & "','" & Text1(5) & "
','" & Text1(8) & "','" & Text1(7) & "','" & Str(DT2.Value) & "','"
& Text1(9) & "','" & Text1(10) & "','" & Text1(11) & "')")
            MsgBox "数据保存成功", 64, "提示信息"
         Else
         End If
           Adodc1.RecordSource = "select * from tbCarInfo
order by carid"
           Adodc1.Refresh
           Call DBGCon
      End If
  With tlbCarInfo
```


shop35833438.taobao.com

```
        .Buttons(1).Enabled = True
        .Buttons(2).Enabled = True
        .Buttons(3).Enabled = True
        .Buttons(4).Enabled = False
        .Buttons(5).Enabled = True
        .Buttons(6).Enabled = True
    End With
  End If
End Sub
Private Sub Edits()    ´修改信息的事件过程
On Error Resume Next
   AdoRs.Open "select * from tbCarDispatch where carnumber=´" + Text1(1) + "´", Cnn, adOpenKeyset
   If AdoRs.RecordCount > 0 Then
      MsgBox "该车已被派遣,不允许修改", 48, "提示信息"
   Else
    c = MsgBox("确认修改信息吗", 33, "提示信息")
     If c = vbOK Then
        Set AdoRs = Cnn.Execute ("UPDATE tbCarInfo SET carid =´" + Text1 (0) + "´,carnumber =´" + Text1 (1) + "´, carclass =´" + cbxCar + "´,drivername =´" + Text1 (2) + "´, birthdate=´" + Str(DT2.Value) + "´,officer=´" _
        + Text1(3) + "´,drivercard=´" + Text1(4) + "´,email=´" + Text1(5) + "´,officernumber=´" + Text1(6) + "´,officerphone=´" + Text1(7) + "´,driverphone=´" + Text1(8) + "´,postcode=´" + Text1(9) + "´,address=´" _
        + Text1 (10) + "´,appendix=´" + Text1 (11) + "´ where carid=´" + Text1(0).Text + "´")
        MsgBox "数据修改成功", 64, "提示信息"
         Adodc1.RecordSource = "select * from tbCarInfo order by carid"
        Adodc1.Refresh
          StrId = Text1(0).Text
          StrTitle = Text1(1).Text
          Call DBGCon
     Else
     End If
    End If
  AdoRs.Close
End Sub
Private Sub LoadFile()
  AdoRs.Open "select * from tbCarInfo", Cnn, adOpenKeyset
   If AdoRs.RecordCount > 0 Then
          Text1(0).Text = AdoRs.Fields("carid")
          Text1(1).Text = AdoRs.Fields("carnumber")
          cbxCar.Text = AdoRs.Fields("carclass")
          Text1(2).Text = AdoRs.Fields("drivername")
          DT2.Value = AdoRs.Fields("birthdate")
          Text1(3).Text = AdoRs.Fields("officer")
          Text1(4).Text = AdoRs.Fields("drivercard")
          Text1(5).Text = AdoRs.Fields("email")
          Text1(6).Text = AdoRs.Fields("officernumber")
          Text1(7).Text = AdoRs.Fields("officerphone")
          Text1(8).Text = AdoRs.Fields("driverphone")
          Text1(9).Text = AdoRs.Fields("postcode")
          Text1(10).Text = AdoRs.Fields("address")
          Text1(11).Text = AdoRs.Fields("appendix")
     End If
   AdoRs.Close
End Sub
Private Sub Deletes()                    ´删除信息
   AdoRs.Open "select * from tbCarDispatch where carnumber=´" + Text1(1) + "´", Cnn, adOpenKeyset
   If AdoRs.RecordCount > 0 Then  ´如果该信息正处于使用´当中,则禁止删除信息
      MsgBox "该车已被派遣,不允许删除", 48, "提示信息"
   Else
   c = MsgBox("确认删除该信息吗", 17, "提示信息")
     If c = vbOK Then
        On Error Resume Next
         Set AdoRs = Cnn.Execute ("Delete tbCarInfo from tbCarInfo where carid=´" + Text1(0).Text + "´")
        Adodc1.Refresh
     End If
     For i = 1 To 15
       Text1(i).Text = ""
     Next i
   End If
 AdoRs.Close
End Sub
Private Sub DBGCon()
    dgrContent.Columns(0).Caption = "编号"
    dgrContent.Columns(1).Caption = "车牌号码"
    dgrContent.Columns(2).Caption = "车辆类型"
    dgrContent.Columns(3).Caption = "司机姓名"
    dgrContent.Columns(4).Caption = "负责人姓名"
    dgrContent.Columns(5).Caption = "负责人电话"
    dgrContent.Columns(6).Caption = "身份证号"
    dgrContent.Columns(7).Caption = "电子邮件"
    dgrContent.Columns(8).Caption = "司机电话"
    dgrContent.Columns(9).Caption = "负责人电话"
    dgrContent.Columns(10).Caption = "出生日期"
    dgrContent.Columns(11).Caption = "邮政编码"
    dgrContent.Columns(12).Caption = "负责人单位"
    dgrContent.Columns(13).Caption = "备注信息"
    Dim i As Integer
    For i = 0 To 13
    Me.dgrContent.Columns(i).Width = 1420
    Next i
End Sub
Private Sub Timer1_Timer()
  On Error Resume Next
    Lbl_Num.Caption = " 当前数据表中共有 " & Adodc1.Recordset.RecordCount & " 条记录"
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
End Sub
Private Sub Form_Initialize()
Call InitCommonControls '设置XP效果
End Sub
Private Sub Form_Resize()
If Me.ScaleHeight > 8295 Or Me.ScaleWidth > 11280 Then
        Me.dgrContent.Height = Me.ScaleHeight - Me.
dgrContent.Top
  Me.dgrContent.Width = Me.ScaleWidth - Me.dgrContent.
Left
End If
End Sub
```

3.5 车辆派遣信息查询 (frmCarDispatch.frm)

车辆派遣信息查询的运行界面如图7所示。



图7 车辆派遣信息查询

```
Dim i As Integer
Dim c
Private Sub dgrContent_Click()
  On Error Resume Next
  If Adodc1.Recordset.RecordCount > 0 Then
    txtCardId.Text = Adodc1.Recordset.Fields("ticketid")
    Text1(0).Text = Adodc1.Recordset.Fields("carnumber")
    Text1(1).Text = Adodc1.Recordset.Fields("leadername")
    Text1(2).Text = Adodc1.Recordset.Fields("drivername")
    Text1(3).Text = Adodc1.Recordset.Fields("reason")
    Text1(4).Text = Adodc1.Recordset.Fields("stuff")
    DT1.Value = Adodc1.Recordset.Fields("outdate")
    Text1(5).Text = Adodc1.Recordset.Fields("dest")
    txtRemark.Text = Adodc1.Recordset.Fields("appendix")
   End If
   For i = 0 To 6
     Text1(i).Locked = False
   Next i
End Sub
Private Sub DT1_KeyDown (KeyCode As Integer, Shift As
Integer)
  If KeyCode = 13 Then
    Text1(4).SetFocus
  End If
End Sub
Private Sub Form_Initialize()
Call InitCommonControls '设置XP效果
End Sub
Private Sub Form_Load()
On Error Resume Next
  Me.Left = (Screen.Width - Me.Width) / 2
  Me.Top = (Screen.Height - Me.Height) / 2
  Call LoadFile
  For i = 1 To 5
    Text1(i).Locked = False
  Next i
 Adodc1.ConnectionString = "Provider=Microsoft.Jet.OLEDB.
4.0;Data Source=" & App.Path & "\data\data.mdb;Jet OLEDB:
Database Password=123456;Persist Security Info=False;"
      Adodc1.RecordSource = "select * from tbCarDispatch
order by ticketid"
  Adodc1.Refresh
  DT1.Value = Date
      Call DBGCon
  With Me.tlbCarDispatch
      .Buttons(1).Enabled = True
      .Buttons(2).Enabled = True
      .Buttons(3).Enabled = True
      .Buttons(4).Enabled = False
      .Buttons(5).Enabled = True
      .Buttons(6).Enabled = True
  End With
End Sub
Private Sub Label9_Click()
End Sub
Private Sub Form_Resize()
If Me.ScaleHeight > 8295 Or Me.ScaleWidth > 11280 Then
    Me.dgrContent.Height = Me.ScaleHeight - Me.
dgrContent.Top
  Me.dgrContent.Width = Me.ScaleWidth - Me.dgrContent.
Left
End If
End Sub
Private Sub tlbCarDispatch_ButtonClick (ByVal Button As
MSComctlLib.Button)
  Select Case Button.Index
    Case 1
With Me.tlbCarDispatch
    .Buttons(1).Enabled = False
    .Buttons(2).Enabled = True
    .Buttons(3).Enabled = True
    .Buttons(4).Enabled = True
    .Buttons(5).Enabled = True
    .Buttons(6).Enabled = True
End With
    For i = 0 To 5    '单击"添加"按钮,锁定文本框控件
      Text1(i).Text = ""
      txtRemark.Text = ""
```


shop35833438.taobao.com

```
        Text1(0).SetFocus
        Text1(i).Locked = False
        txtCardId.Locked = False
        txtRemark.Locked = False
      Next i
       AdoRs.Open "select * from tbCarDispatch order by
ticketid", Cnn, adOpenKeyset
        If AdoRs.RecordCount > 0 Then
          AdoRs.MoveLast
          StrNum = Mid (AdoRs.Fields ("ticketid"), 2, Len
(AdoRs.Fields("ticketid")))
          Call CovertNum      '位数转换函数
          txtCardId.Text = "A" & StrIn
        Else
          txtCardId.Text = "A0001"
        End If
      AdoRs.Close
  Case 2                      '删除信息
    Call Deletes
    Call DBGCon
  Case 3                      '修改信息
    Call Edits
    Call DBGCon
  Case 4                      '保存信息
   Call Saves
    Call DBGCon
  Case 5
 With Me.tlbCarDispatch
      .Buttons(1).Enabled = True
      .Buttons(2).Enabled = True
      .Buttons(3).Enabled = True
      .Buttons(4).Enabled = False
      .Buttons(5).Enabled = True
      .Buttons(6).Enabled = True
End With
     For i = 0 To 5
       Text1(i).Text = ""
       txtCardId.Text = ""
       txtRemark.Text = ""
       txtCardId.SetFocus
       Text1(i).Locked = True
       txtCardId.Locked = True
       txtRemark.Locked = True
     Next i
     Adodc1.RecordSource = "select * from tbCarDispatch
order by ticketid"
     Adodc1.Refresh
     Call DBGCon
  Case 6                      '打印信息
    On Error Resume Next
      MsgBox "打印失败,请联系作者!"
  Case 7
      Unload Me
    Case 9
    frmCarQuery.Show 1
  End Select
End Sub
Private Sub Text1_KeyDown (Index As Integer, KeyCode As
Integer, Shift As Integer)
  On Error Resume Next
  If KeyCode = 13 Then
    If Index = 0 Then
      Exit Sub
    End If
  End If
  If KeyCode = 13 Or KeyCode = 40 Then
    If Index = 3 Then
      DT1.SetFocus
      Exit Sub
    End If
  End If
  If Index = 5 Then
    If KeyCode = 13 Or KeyCode = 40 Then
      txtRemark.SetFocus
    End If
  End If
    If Index = 4 Then
      If KeyCode = 38 Then
        DT1.SetFocus
        Exit Sub
      End If
    End If
  If Index < 6 And KeyCode = 38 Then Text1 (Index - 1).
SetFocus
  If Index >= 0 And KeyCode = 40 Then Text1 (Index + 1).
SetFocus
  If Index >= 0 And KeyCode = 13 Then Text1 (Index + 1).
SetFocus
End Sub
Private Sub Saves()    '保存信息的事件过程
  If Text1(0).Text = "" Or Text1(3).Text = "" Then
    MsgBox "重要信息不能为空值", 48, "提示信息"
  Else
      AdoRs.Open "select * from tbCarDispatch where
carnumber='" + Text1(0).Text + "'", Cnn, adOpenKeyset
      If AdoRs.RecordCount > 0 Then
        MsgBox "该车辆已经被派出", 48, "提示信息"
        AdoRs.Close
      Else
        AdoRs.Close
       c = MsgBox("确认保存信息吗", 33, "提示信息")
        If c = vbOK Then
          Set AdoRs = Cnn.Execute ("insert into
tbCarDispatch values('" & txtCardId & "','" & Text1(0) _
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
  & "','" & Text1(1) & "','" & Text1(2) & "','" & Text1(3) & "','" _
          & Str(DT1.Value) & "','" & Text1(4) & "','" & Text1
(5) & "','" & txtRemark & "')")
          MsgBox "数据保存成功", 64, "提示信息"
        Else
        End If
          Adodc1.RecordSource = "select * from
tbCarDispatch order by ticketid"
          Adodc1.Refresh
          Call DBGCon
      End If
  With Me.tlbCarDispatch
      .Buttons(1).Enabled = True
      .Buttons(2).Enabled = True
      .Buttons(3).Enabled = True
      .Buttons(4).Enabled = False
      .Buttons(5).Enabled = True
      .Buttons(6).Enabled = True
  End With
  End If
End Sub
Private Sub Edits()    '修改信息的事件过程
   c = MsgBox("确认修改信息吗", 33, "提示信息")
    If c = vbOK Then
      Set AdoRs = Cnn.Execute ("UPDATE tbCarDispatch
SET ticketid='" + txtCardId + "',carnumber='" + Text1(0) + "',
leadername='" + Text1(1) + "',drivername='" + Text1(2) + "',
outdate='" _
      + Str(DT1.Value) + "',reason='" + Text1(3) + "',stuff='"
+ Text1 (4) + "',dest ='" + Text1 (5) + "',appendix ='" +
txtRemark + "' where ticketid='" + txtCardId.Text + "'")
      MsgBox "数据修改成功", 64, "提示信息"
      Adodc1.RecordSource = "select * from tbCarDispatch
order by ticketid"
      Adodc1.Refresh
       StrId = txtCardId.Text
       StrTitle = Text1(0).Text
      Call DBGCon
    Else
    AdoRs.Close
    End If
End Sub
Private Sub LoadFile()
    AdoRs.Open "select * from tbCarDispatch", Cnn,
adOpenKeyset
    If AdoRs.RecordCount > 0 Then
      txtCardId.Text = AdoRs.Fields("ticketid")
      Text1(0).Text = AdoRs.Fields("carnumber")
      Text1(1).Text = AdoRs.Fields("leadername")
      Text1(2).Text = AdoRs.Fields("drivername")
      Text1(3).Text = AdoRs.Fields("reason")
      Text1(4).Text = AdoRs.Fields("stuff")
      DT1.Value = AdoRs.Fields("outdate")
      Text1(5).Text = AdoRs.Fields("dest")
      txtRemark.Text = AdoRs.Fields("appendix")
    End If
  AdoRs.Close
End Sub
Private Sub Deletes()                    '删除信息
  c = MsgBox("确认删除该信息吗", 17, "提示信息")
    If c = vbOK Then
      On Error Resume Next
      Set AdoRs = Cnn.Execute("Delete tbCarDispatch from
tbCarDispatch where ticketid='" + txtCardId.Text + "'")
      Adodc1.Refresh
    End If
    For i = 0 To 5
      Text1(i).Text = ""
    Next i
     txtRemark.Text = ""
End Sub
Private Sub DBGCon()
   dgrContent.Columns(0).Caption = "派车单编号"
   dgrContent.Columns(1).Caption = "车牌号码"
   dgrContent.Columns(2).Caption = "带车干部"
   dgrContent.Columns(3).Caption = "司机姓名"
   dgrContent.Columns(4).Caption = "执行任务"
   dgrContent.Columns(5).Caption = "执行任务日期"
   dgrContent.Columns(6).Caption = "携带装备器材"
   dgrContent.Columns(7).Caption = "目的地"
   dgrContent.Columns(8).Caption = "备注"
   Dim i As Integer
   For i = 0 To 8
   Me.dgrContent.Columns(i).Width = 1420
   Next i
End Sub
Private Sub txtRemark_KeyDown (KeyCode As Integer, Shift
As Integer)
  If KeyCode = 38 Then
    Text1(5).SetFocus
  ElseIf KeyCode = 13 Then
    Call Saves
  End If
End Sub
Private Sub Timer1_Timer()
  On Error Resume Next
    Lbl_Num.Caption = " 当前数据表中共有 " & Adodc1.
Recordset.RecordCount & " 条记录"
End Sub
Private Sub txtCardId_KeyDown (KeyCode As Integer, Shift
As Integer)
  If KeyCode = 13 Then
    Adodc1.RecordSource = "select * from tbCarDispatch
```

（下转第 62 页）

LAICAR.COM
shop35833438.taobao.com

串口通信新模型的研究与 C# 实现

郑 武 肖宝森

摘 要：借鉴 I/O 完成端口模型（IOCP）的核心思想，建立了一个适用于协议性串口通信的新模型。该模型提出了一个协议性串口通信的最小单位——“通信元”，将“通信元”提交到事件队列线程中自动处理，简化了处理过程并提高了运行效率。同时，基于该模型用 C# 设计出了一个协议性串口通信模块，以封装通信底层的细节，从而提供方便的通信接口。将该模块应用于工业造气炉监控组态软件设计中，实际运行表明：该模型具有简单的接口和稳定、高效的运行机制。

关键词：IOCP 模型；串口通信；通信元

1 引言

串口即串行通信接口，在工业控制领域，它是连接计算机和外部串行设备的常用数据传输通道。大量的智能仪器或数据采集模块都带有串口，通过相应的通信协议和串口通信程序，可以将测试、采集的数据传输到上位工控机，以满足各种形式后续数据处理的要求。同时上位监控系统也会根据系统运行情况通过串口向监控设备发送指令，调节设备的运行参数等使得生产过程平稳进行。串口通信具有稳定可靠、成本低廉、软件易实现等优点，是当前工业控制行业运用比较广泛的通信方式。

传统的串口编程要综合考虑底层通信细节和串口通信协议。这一方面导致调试串口的时候要对这两种情况进行分析，增加了调试的困难；另一方面二者深度耦合，当串口通信协议改变的时候整个通信程序要重写，不符合现代编程的模块化思想。鉴于此，参考 I/O 完成端口模型（IOCP）建立了一个适用于协议性串口通信的新模型，将底层通信的细节进行封装并和通信协议分离，很好地解决了上述问题。

2 串口通信新模型

2.1 完成端口简介

IOCP 是性能最好的一种 I/O 模型。它是应用程序使用线程池处理异步 I/O 请求的一种机制。通过将完成端口与任意 I/O 句柄（文件或 Socket 等）关联，用户可以通过完成端口异步地获取并处理 I/O 的结果。

如图 1 所示，在完成端口上建立几个并行的工作线程，它们为到达完成端口的 I/O 任务提供服务。当有 I/O 任务到达时，如果有可用的工作线程，则激活该线程，如果没有可用工作线程，则将 I/O 任务加入到请求队列。该队列采用先进先出（FIFO）的策略，保证这些请求得到公平的服务。工作线程的建立和请求队列的 FIFO 策略，减少了 CPU 在不同线程间切换的次数，降低线程上下文切换所造成的开销[2]。

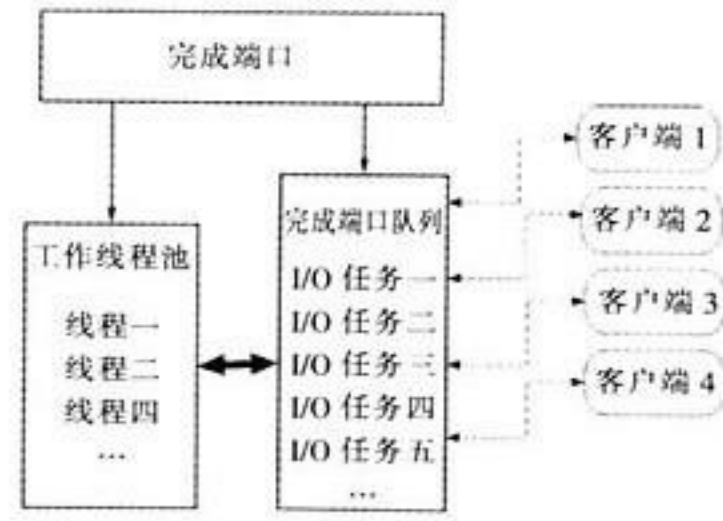


图 1 I/O 完成端口模型

2.2 串口通信新模型

借鉴 IOCP 的核心思想，模型提出了一个协议性串口通信的最小单位——“通信元”。通信元包含发送数据缓存区、接收数据缓存区、接收数据长度、超时时间间隔、超时处理函数和事件处理函数 6 个部分。其中数据缓存区用来存放发送数据和接收数据，超时时间间隔用来设置串口通信超时时间，超时处理函数对超时情况进行处理，而事件处理函数则对接收到的数据进行处理。

模型维护一个通信元队列，当进行串口通信时先定义好通信元然后提交到通信元队列，队列根据先进先出的原则进行处理。当某个通信元出队时，模型先发送该通信元发送数据缓存区的数据，接着根据接收数据长度对串口数据进行接收，如果超时则调用超时处理函数，否则进行接收事件处理。通过该模型，将每次串口通信的过程简化成一个通信元的处理过程，用户只需要定义好通信元提交到队列中，模型即自动处理底层通信的细节，简化了处理流程并提高了运行效率。

来卡网出品
LAICAR.COM
shop35833438.taobao.com

3 C# 实现

C# 是微软推出的一种基于.NET 框架的、面向对象的高级编程语言。C# 由 C 语言和 C++派生而来，继承了其强大的性能，同时又以.NET 框架类库作为基础，拥有类似 Visual Basic 的快速开发能力。C# 自.NET 2.0 开始提供了 SerialPort 类实现对串口通信的原生支持，本模型在这个类基础上进行编程实现。

首先是本模型最重要的通信元，代码如下：

```
public class SPI_EventData
  {
     public Byte[] txd_MSG;   //发送数据缓存区
     public int rxd_Len;      //接收数据长度
     public Byte[] rxd_MSG = new byte[512]; //接收数
//据缓存区

     public int timeOut;    //超时时长
       public   delegate   void   func   (SPI_EventData
eventData);    //事件函数委托
     public func eventFunc;      //事件处理函数
     public func overTimerFunc;  //超时处理函数
  };
```

其次是通信元队列处理，这里定义了一个线程专门对通信元队列进行处理。线程有两个阻塞信号，一个用来判断队列是否为空，一个用来判断当前通信元是否处理完毕。代码如下：

```
private void eventEngineFunc()
 {
    while (true)
    {
       eventSingal.WaitOne();  //等待事件信号
       eventFinish.WaitOne();  //等待事件完成信号
       if (eventQueue.Count ! = 0)
       {
          eventFinish.Reset();   //事件完成信号复位
          currentEvent = outQueue(); //事件出队
          sendMSG();   //串口发送数据
       }
       else
       {
       //事件队列空事件信号复位,释放队列内存
          eventSingal.Reset();
          eventQueue.TrimToSize();
       }
    }
 }
```

最后是串口接收函数，模型自动根据通信元中接收数据长度对串口数据进行接收。代码如下：

```
    private   void   receiveMSG   (object   sender,
SerialDataReceivedEventArgs e)
{
   try
   {
                //读系统串口缓存到接收缓冲区
               currentLenMSG  +=  sp.Read (currentEvent.
rxd_MSG, currentLenMSG, currentEvent.rxd_Len – currentL
enMSG);
   }
      catch
      {
 sp.DataReceived –= rxd_Event; //释放串口接收事件
         overTimer.Stop();  //停止超时检测
         try
         {
            currentEvent.overTimerFunc(currentEvent);
//超时处理
         }
         catch
         {
               MessageBox.Show     (Language.errStr6,
Language.attention);
         }
         sp.DiscardInBuffer();  //清空接收区防止再次接收
   eventFinish.Set();   //发送事件完成信号
         return;
      }
        if    (currentLenMSG   <   currentEvent.rxd_Len)
//如果没读完继续读取
      {
         return;
      }
     sp.DataReceived –= rxd_Event;  //释放串口接收事件
       if (overTimer.isOverTime)  //如果已经超时,返回交给
//超时函数处理
      {
         return;
      }
      overTimer.Stop();      //停止超时检测

      try
      {
         currentEvent.eventFunc(currentEvent); //事件处理
      }
      catch
      {
MessageBox.Show(Language.errStr7, Language.attention);
      }
      sp.DiscardInBuffer(); //清空接收区防止再次接收
      eventFinish.Set();   //发送事件完成信号
   }
```

其他包括串口发送函数、通信元入队、出队函数、定时器

（下转第 66 页）

shop35833438.taobao.com

移动平台下天气信息查询软件的开发

王浩亮

摘 要：在对当前智能手机系统分析、研究的基础上，提出了基于 Android 平台的天气信息查询软件设计方案，采用 XML 和 JSON 两种数据载体格式，通过 WebService 获取天气信息和地理位置信息，将信息进行分析处理，保存在 sqlite 数据库中，利用 Achartengine 绘图技术生成图表呈现给用户。

关键词：天气信息查询； Android 平台； WebService 服务； Achartengine 绘图技术

1 引言

随着智能手机的快速普及，智能手机的应用软件及其需要的服务将有广阔的发展前景。在如今智能手机系统群雄纷争的时代，Google 推出 Android 开源智能手机操作系统，它采用 Linux 内核，系统的开源性使其具有良好的拓展性。Android 最大特点的是其开放性体系架构不仅具有非常好的开发调试环境，而且还支持各种可扩展的用户体验，包括丰富的图形组件、多媒体支持功能以及强大的浏览器。文中在分析讨论 Android 手机软件开发技术原理的基础上，开发出能为用户提供更好的基于 Android 的天气信息查询软件。

天气信息查询软件是 Android 移动手机端必备的手机程序之一，在综合了以往的在线天气系统后，研究课题增加了一些比较实用的模块。例如：本天气信息查询软件的未来天气分析模块，为用户提供了未来 4 天内的天气走向图，让用户更直观地了解到未来的天气变化情况。公众能够在第一时间方便且快捷地获取最新的气象预报（预警）信息，以便提前预防，把气象灾害造成的损失降到最低，方便用户出行。

2 技术原理

2.1 Android 平台

Android 是 Google 开发的基于 Linux 平台的开源手机操作系统的名称，该平台由操作系统、中间件、用户界面和应用软件组成，号称是首个为移动终端打造的真正开放和完整的移动软件。Android 平台有一个运行在 Linux 内核之上的 Dalvik 虚拟机。Dalvik 虚拟机在保证 API 兼容的同时，针对移动手机进行了大幅优化，占用资源更小、运行效率更高。应用开发者能够使用 Java 编程语言，基于 Android SDK 平台开发大量的增值应用。

2.2 SAX 解析器

SAX 即是：Simple API for XML。SAX 是一个解析速度快并且占用内存少的 XML 解析器，非常适合用于 Android 等移动设备。SAX 解析 XML 文件采用的是事件驱动，也就是说，它并不需要解析完整个文档，在按内容顺序解析文档的过程中，SAX 会判断当前读到的字符是否为 XML 语法中的某部分，如果符合就会触发事件。所谓事件，其实就是一些回调（callback）方法，这些方法（事件）定义在 ContentHandler 接口。

2.3 Json 数据格式

XML 这种用于表示客户端与服务器间数据交换有效负载的格式，几乎已经成了 Web Services 的同义词。然而，由于 Ajax 和 REST 技术的出现影响了应用程序架构，这迫使人们开始寻求 XML 的替代品，如：JavaScript Object Notation（JSON）。

JSON (JavaScript Object Notation) 是一种轻量级的数据交换格式。它基于 JavaScript 的一个子集。JSON 采用完全独立于语言的文本格式，但是也使用了类似于 C 语言家族的习惯（包括 C, C++, C#, Java, JavaScript, Perl, Python 等）。这些特性使 JSON 成为理想的数据交换语言。易于人阅读和编写，同时也易于机器解析和生成。

2.4 Achartengine 开源绘图库

Achartengine 是为 Android 应用而设计的开源绘图工具库，可用于绘制多种图表。Achartengine 支持的图表类型：折线图、区域图、散点图、时间图、柱状图、饼状图、气泡式图表、环形图、高低交替图。以上每种图表都可以包含多个序列，可以将 X 轴以水平或垂直的形式显示，也支持很多自定义的特性。另外，图表可以作为 view 构建，也可以作为 Intent 构建——这样可以被用于启动一个 Activity。

3 功能描述

整个应用程序应划分为 5 个部分，分别是用户界面、后台服务、城市管理、数据库适配器和天气趋势展示。

（1）后台服务是整个应用程序的核心，主要是“天气信息

来卡网出品
LAICAR.COM
shop35833438.taobao.com

数据获取模块”，负责按照用户设定的时间进行周期性地从WebService中获取天气信息，这里我们就Google提供的天气信息为例，并把数据存入数据库。后台服务在程序启动时就由oncreate() 方法启动，也可由用户界面通过startService() 进行启动，启动后的后台服务将一直保持运行状态。

(2) 用户界面从数据库中获取天气信息，而没有直接通过网络访问Google的天气数据。一方面可以避免因网络通信不畅造成界面失去响应；另一方面，后台服务可以及时更新数据，以保证数据的准确性。

(3) 城市管理可让用户添加用户所需的城市到数据库，并可以通过移动定位功能进行用户定位，把定位得到的城市添加到数据库。

(4) 数据库适配器封装了所有对SQLite数据库操作的方法，用户界面和后台服务会调用它来实现数据库操作。

(5) 天气趋势图通过折线图更加直观地为用户展现未来几天的天气走向，包括日间温度、夜间温度。

功能结构图如图1所示。



图1 系统功能结构

4 系统实现

4.1 系统主界面效果

主界面显示当天的天气具体情况和未来3天的天气信息，效果如图2所示。



图2 系统运行主界面

4.2 天气信息的获取

把城市名称提交到Google weather api，然后根据其所反馈的XML格式的天气信息，用SAX解析器来解析这些XML，提取有用的天气信息以beijing为例。访问Google weather api的URL为：“http://www.google.com/ig/api?hl=zh-cn&weather=beijing”。得到了确切的XML数据后，用SAX解析器对这个XML进行解析，构造SAX解析器和XML适配器的代码如下，其中GoogleWeatherHandler类为继承了DefaultHandler用adapter模式实现了SAX解析器的常用接口：

```
SAXParserFactory spf = SAXParserFactory.newInstance();
SAXParser sp = spf.newSAXParser();
XMLReader xr = sp.getXMLReader();
GoogleWeatherHandler gwh = new GoogleWeatherHandler();
xr.setContentHandler(gwh);
```

在完成以上的构造后，还需要用输入流来读取XML，并传递给解析器，代码如下：

```
InputStreamReader isr = new InputStreamReader (url.
openStream(), "GBK");
InputSource is = new InputSource(isr);
xr.parse(is);
```

最后，把解析器所读取的数据赋给已经建好的天气对象类WeatherSet，并把数据存入数据库，代码如下：

```
WeatherSet ws = gwh.getMyWeatherSet();
ws.insertOrUpdate(CityWeather.this, city);
```

4.3 城市管理模块

主要实现增删用户城市。用户可以把城市名的中文名或拼音填入输入框进行添加，在“我的城市”列表中，可以点击城市名进行删除城市操作。添加城市功能主要代码：

```
dbHelper = new DatabaseHelper(CityEdit.this,"
weatherdb.db",null,1);
db = dbHelper.getWritableDatabase();
ContentValues values = new ContentValues();
values.put("cityName", cityName);
db.insert("T_city", null, values);
```

显示列表主要代码：

```
ArrayList<HashMap<String, Object>> listItem
= new ArrayList<HashMap<String, Object>>();
if(cur.getCount() ! = 0){
    if( cur.moveToFirst() ){
        do{
HashMap <String, Object > map = new
HashMap<String, Object>();
map.put("cityName", cur.getString(1));
listItem.add(map);
    }while( cur.moveToNext());
}
```

4.4 温度趋势效果

使用aChartEngine开源绘图api进行本功能的折线图的绘制，效果如图3所示。

LAICAR.COM
shop35833438.taobao.com



图 3 温度趋势图

实现流程：

（1）定义曲线名称和坐标点，x1,x2,y1,y2 为坐标数据，类型为 double[]，数据从数据库读取：

```
String[] titles = new String[]{"白天","夜间"};
    List x = new ArrayList(); //点集的 x 坐标
    List y = new ArrayList(); //点集的 y 坐标
    x.add(x1);
    x.add(x2);
    y.add(y1);
    y.add(y2);
```

（2）创建 XYSeries 对象，用于提供绘制的点集合的数据：

```
XYSeries series = new XYSeries(titles[i]); // 根据每条线的名
//称创建
series .add (xV[k], yV[k]);   //将所有的坐标加入这个集合
// XYMultipleSeriesDataset 对象用于保存点集数据，包括每条
//曲线的 X,Y 坐标
XYMultipleSeriesDataset dataset = new XYMultipleSeries
Dataset();
dataset .addSeries (series);
```

定义每条曲线的颜色和每条曲线的点的形状：

```
int [] colors = new int [] {Color. BLUE ,Color. GREEN };
PointStyle [] styles =
new PointStyle [] { PointStyle . CIRCLE , PointStyle . DIAMOND
};
```

曲线图的格式，包括颜色，值的范围，点和线的形状等等，都封装在 XYSeriesRender 对象中，将 XYSeriesRender 对象封装在 XYMultipleSeriesRenderer 对象中，并设置折线图名称，坐标轴的名称，坐标轴的起点重点，坐标轴的颜色，坐标轴上数字的颜色：

```
XYSeriesRenderer r = new XYSeriesRenderer();
 r.setColor(colors[i]);
 r.setPointStyle(styles[i]);
 r.setFillPoints(fill);
XYMultipleSeriesRenderer renderer = new XYMultipleSeries
Renderer();
renderer.addSeriesRenderer(r);
setChartSettings(renderer, "温度趋势图", "日期", "温度", x1[0]-
1, x1[3]+1, y2[0]-10, y1[3]+10, Color.WHITE, Color.WHITE);
//得到折线图
View chart = ChartFactory.getLineChartView (this, dataset,
renderer);
return chart;
```

5 结语

在介绍基于 Android 平台的天气查询软件实现技术原理的基础上，给出了 Android 用户界面设计、获取并解析城市列表数据的一种方法。另外软件的天气信息获取也可以通过新浪天气、腾讯天气等 WebService 服务，读者还可以在软件功能基础上穿衣指数、晨练指数、洗车指数和天气报警等功能。

参考文献

[1] 刘正，陈强. 基于 Android 的 GPS 应用中的研究与开发 [J]. 中国西部科技，2010.

[2] 郝玉龙. Android 程序设计基础 [M]. 北京：清华大学出版社，2011，1.

（收稿日期：2013-01-06）

（上接第 57 页）

```
where ticketid='" + txtCardId.Text + "'"
    Adodc1.Refresh
    If Adodc1.Recordset.RecordCount > 0 Then
      txtCardId.Text = Adodc1.Recordset.Fields("ticketid")
      Text1(0).Text = Adodc1.Recordset.Fields("carnumber")
      Text1(1).Text = Adodc1.Recordset.Fields("leadername")
      Text1(2).Text = Adodc1.Recordset.Fields("drivername")
      Text1(3).Text = Adodc1.Recordset.Fields("reason")
      Text1(4).Text = Adodc1.Recordset.Fields("stuff")
      DT1.Value = Adodc1.Recordset.Fields("outdate")
      Text1(5).Text = Adodc1.Recordset.Fields("dest")
      txtRemark.Text = Adodc1.Recordset.Fields("appendix")
      End If
      Call DBGCon
    End If
End Sub
```

4 结语

系统小巧精练，使用简单，功能较强大，较好地满足了值班员日常工作的实际需求，提高了值班信息的查询效率。当然本系统还有待于进一步的改进，比如，可适当加入视频监控、电话录音、警报发放、数据备份等功能，上文仅仅是抛砖引玉，希望读者能开发出功能更为实用的值班室信息查询系统。

（收稿日期：2013-05-06）

来卡网出品
LAICAR.COM
shop35833438.taobao.com

Android 利用 SOAP 协议调用天气预报服务

庞国明

摘　要：使用 Android 设计技术开发了一种运用在 Android 系统上的手机天气预报软件系统，系统通过选择城市来获得天气、风向、温度等信息。手机的天气预报系统可以使用户对于各地的天气情况实时掌握，极大地方便了用户的出行和行程安排，避免了不必要的麻烦，具有很强的实用性。

关键词：Android 系统平台；SOAP 协议； kSOAP2 软件；天气预报系统；

1　引言

在科技日益发达的今天，社会竞争越来越激烈，这要求人们不断地摄取信息，同时手机发展到现在已经成为人们日常使用的通读工具，手机的功能也日趋完善。我国的通读移动通信功能同样得到了很大的提高。手机软件可以安装在手机上的软件，完善原始系统的不足，并使手机更加个性化。随着科技的发展，现在手机的功能也越来越多，越来越强大。经济发展和社会进步，人们的出行更加频繁，对于天气的关注更加强烈，基于这种原因我们开发了一种运用在 Android 系统上的手机天气预报软件系统，本系统通过选择城市来获得天气、风向、温度等信息。基于手机的天气预报系统软件可以使用户对于各地的天气情况的实时掌握，极大地方便了用户的出行和行程安排，避免了不必要的麻烦。Android 手机系统可以准确预测包括：时间、温度、风向等，并显示在手机界面上，方便准确地了解天气信息。

2　体系结构

2.1　Android SDK 架构

Android 平台由操作系统、中间件、用户界面和应用软件组成（如图 1 所示）。它采用软件堆层的架构，主要分为 3 部分。底层以 Linux 内核工作为基础，由 C 语言开发，只提供基本功能；中间层包括函数库 Library 和虚拟机 Virtual Machine，由 C++ 开发。最上层是各种应用软件，包括通话程序、短信程序等，应用软件则由各公司自行开发，以 Java 作为编写程序的一部分。

第一层，应用程序层，该层提供一些核心应用程序包，例如电子邮件、短信、日历、地图、浏览器和联系人管理等。同时，开发者可以利用 Java 语言设计和编写属于自己的应用程序，而这些程序与那些核心应用程序彼此平等、友好共处。

第二层，应用程序框架层，该层是 Android 应用开发的基础，开发人员大部分情况是在和它打交道。应用程序框架层包括活动管理器、窗口管理器、内容提供者、视图系统、包管理器、电话管理器、资源管理器、位置管理器、通知管理器和 XMPP 服务 10 个部分。

第三层，系统库和 Android 运行时，系统库包括 9 个子系统，分别是图层管理、媒体库、SQLite、OpenGLEState、FreeType、WebKit、SGL、SSL 和 libc。Android 运行时包括核心库和 Dalvik 虚拟机，前者既兼容了大多数 Java 语言所需要调用的功能函数，又包括了 Android 的核心库，比如 android.os、android.net、android.media 等等。后者是一种基于寄存器的 java 虚拟机，Dalvik 虚拟机主要是完成对生命周期的管理、堆栈的管理、线程的管理、安全和异常的管理以及垃圾回收等重要功能。

第四层，Linux 内核，Android 核心系统服务依赖于 Linux2.6 内核，如安全性、内存管理、进程管理、网络协议栈和驱动模型。Linux 内核也是作为硬件与软件栈的抽象层驱动：显示驱动、摄像头驱动、键盘驱动、WiFi 驱动、Audio 驱动、Flash 内存驱动、Binder（IPC）驱动、电源管理等。

2.2　开发环境

操作系统：Microsoft Windows 7。



图 1　Android sdk 架构

IDE: Ecplise　Latefrom Version 3.3.1.1。

开发包：Android SDK。

模拟机：Android2.1 或 Android2.3 等。

3　需求分析

3.1　调用天气预报服务设计规划

基于 Android 平台的智能手机访问服务器的主要技术是数

LAICAR.COM
shop35833438.taobao.com

据交互方法。基于 Android 操作系统的天气实况预报系统设计的核心是通过调用通信协议 SOAP（Simple Object AccessProtocol）的接口，从 Web Service 提供商中提取天气预报的数据信息，为客户端服务（如图 2 所示）。其基本功能是当用户运行天气预报系统程序时，在手机屏幕上显示出中国的城市名称，用户单击任意一个城市名，可获得该城市的天气实况预报信息。用户也可以输入城市名称，点击查询获得天气预报信息。具体如下：

（1）能够查询数据库存中城市的天气情况。

（2）系统能通过手机查看并显示天气数据。

（3）系统必须具有安全性，可靠性，独立性。

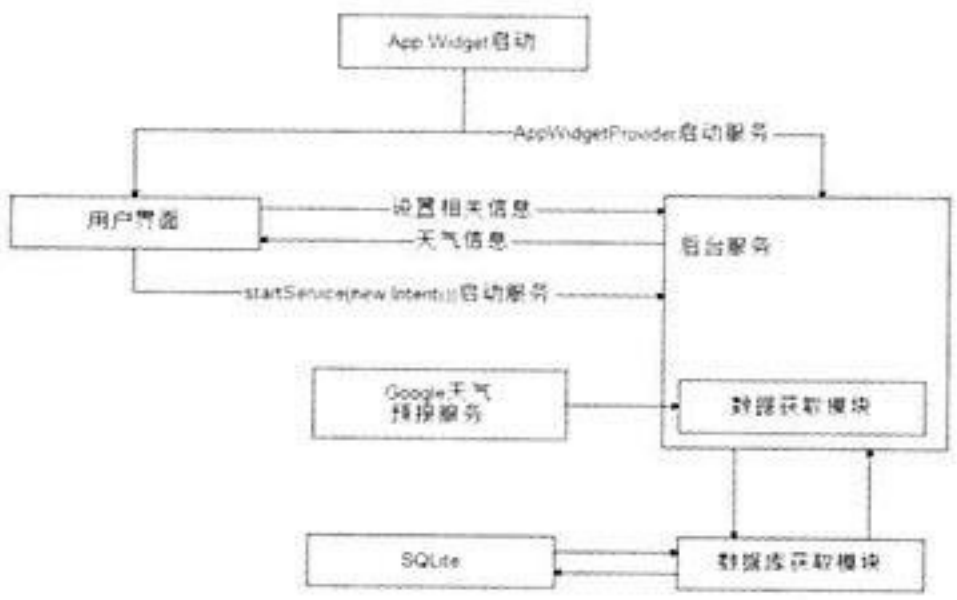


图 2 软件服务设计规划

3.2 性能

（1）由于项目调用 ksoap2.jar 开发包，所以需要在 android 项目中引入 ksoap2.jar。这个开发包是专门以基于 SOAP 协议进行交互的支持包，它可以让 android 项目与 Web Service 服务器进行数据交互。

（2）屏幕显示内容的大小要取决于真实设备的屏幕大小，对于较小的屏幕可能显示的内容会少一部分。

（3）项目本身是 eclipse 环境下的 android 项目，需要在已经添加了 android 插件的 eclipse 中以工程文件形式打开。

3.3 可靠性与可用性

本项目在 motorola defy+ 手机上测试可以正常运行，因此在 android2.1 或更高版本的 android 真实手机上均可运行；在模拟机方面，系统版本为 android2.1 或者更高的版本并保证系统剩余内存为 1M 的模拟机上都可正常运行（如图 3 所示）。

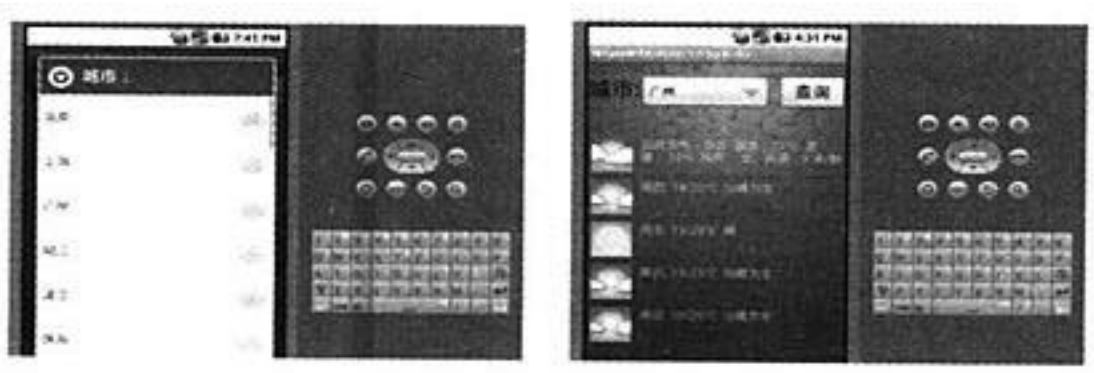


图 3 软件运行效果

4 系统实现

基于 Android 平台的智能手机访问服务器的主要技术是数据交互方法。基于 Android 操作系统的天气实况预报系统设计的核心是通过调用通信协议 SOAP（Simple Object Access Protocol）的接口，从 Web Service 提供商中提取天气预报的数据信息，为客户端服务。其基本功能是当用户运行天气预报系统程序时，在手机屏幕上显示出中国的城市名称，用户单击任意一个城市名，可获得该城市的天气实况预报信息。用户也可以输入城市名称，点击查询获得天气预报信息。

4.1 Android 平台的用户界面

Android 提供的可变化的用户界面（UI）开发模块是基于 XML 文件的。这些 XML 文件放在工程/res/layout 下面。这个目录可包含所有应用程序所需的非码部分，比如图片、字符串、xml 文件。当要使用到这些资源时，在代码目录中打开 R.java 文件即可。在 XML 文件里编辑界面的代码，实现起来不仅方便，使用时也会更加灵活。

在 Android 程序设计中要用到一些基本的 Android UI 元素，通过使用 Views、View Groups 和 layouts 可为 Activity 创建功能性的、富有直观力的 UI.通常是使用 Android SDK 中提供的一些控件，进行布局、扩展和定制这些控件，并使用 ViewGroups 去组合 Views，创建由相互作用的子控件组成的原子的、重复利用的 UI 元素。也可以创建自己的 Views，来实现显示数据和与用户交互的新途径；或使用一些继承自 View Group 的 Layout 管理器来组织 Android UI 中的单个元素到屏幕上。

在一个 Android 应用中，用户界面由 View 和 View Group 对象构建。View 与 View Group 有很多种类，而它们都是 View 类的子类。View 对象是 Android 平台中用户界面的基本单元。View 类是 widgets（工具）类的父类，它们提供了诸如文本输入框和按钮之类的 UI 对象的完整实现。View Group 类是 Layouts（布局）类的父类，它们提供了诸如流式布局、表格布局以及相对布局之类的布局架构。

View 对象是一个数据体，它的属性存储了用于屏幕上一块矩形区域的布局参数及内容。并负责它所辖的这个矩形区域之中所有测量、布局、焦点转换、卷动以及按键/触摸手势的处理。作为一个用户界面对象，View 同时也担任着用户交互关键点以及交互事件接受者的角色。天气实况预报系统程序功能比较单一，其 UI 也相对比较简单，只要有一个输入框、一个查询按钮以及一个显示所有中国城市信息的列表即可。为了建立 Android 平台的用户界面，首先要在 Package Explorer 窗口中展开 Layout 后新建。xml 文件用来对窗口界面进行布局，主要有系统运行的主配置文件 main.xml 和 result.xml.

Layout 是一类特殊的 View Group 控件，它们本身没有任何可显示内容，存在的惟一原因就是其中的内部结构，能够更

来卡网出品
LAICAR.COM
shop35833438.taobao.com

好地摆放它的子控件。比如 Linearlayout，可将子控件按水平或垂直方向按顺序排列下去；Tablelayout，可以将子控件按照表格的形式，一枚枚放置好；Relativelayout 更灵活，可以设定各个控件之间的对齐和排列关系，适合定制复杂的界面。有了 Layout 的存在，控件和控件之间不再是割裂地存在，而是更有机地结合在一起，设定起来也更为方便。在 main.xml 文件中主要是添加一些界面的布局设置，例如如下代码：

```
<LinearLayout
android:orientation="horizontal"
android:layout_width="fill_parent"
android:layout_height="wrap_content">
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/cityText"/>
<Button android:id="@+id/searchBtn"
<Button android:id="@+id/searchBtn"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/searchBtn"/>
</LinearLayout>
```

上述代码生成了一个 Linearlayout,定义了一个 EditText 和 Button，并且设置了相关的参数。在 Android 中，控件最重要的大小属性是 width/height，开发者可以指明控件的大小、控件的宽、高以及在屏幕中的显示位置等，可以设定成为 fill_parent 和 wrap_content。另外，还可以设置相应的文本信息，并可通过 android:text=" @string/cityText" />进行引用(string 文件存放在工程的 res/values 中)。按照这种方法，还可以依据需要定义相应的布局控件，譬如定义用来呈现天气实况信息的 UI 等。

4.2 创建中国城市列表数据

根据天气预报系统功能需求分析，系统启动后，应在主界面上呈现出中国国内所有城市的列表数据，因此需要创建中国城市列表数据，包括获取、解析城市列表数据，并在用户界面上呈现出来。为此，需要解决的问题是手机终端从 Web 服务器获取了所有类型的数据之后，将这些数据进一步交给 Android 手机终端 View 组件，在手机界面上显示给用户。

4.3 获取城市列表数据

Android 没有提供 Web Service 的组件库，但可以通过修改在 PC 或 J2ME 上使用的开源 Web Service Java 库作为 Android Web Servic 库使用。kSOAP2（http://ksoap2.sourceforge.net）是一个在智能手机上的 SOAP Web Service 客户端包，可用于资源受限制的 Java 环境，如 Applets 或 J2ME 应用程序。在 GoogleCode 上有一个项目 ksoap2-android（http://code.google.com/p/ksoap2-android/）可用在 Android 平台上实现轻量级的 SOAP 库，即使用 ksoap2-android 的 API 来调用远端 Web Service 的服务。

因此需要在 CitiesWeatherForecast.java 的 onCreate（）中进行数据获取的初始化工作。通过定义一个方法public List<String> getAllCitiesNAMEs（）可获得中国城市名称列表。也就是说，采用它提供的获得全球各个国家城市的方法 GetCitiesByCountry,通过 http://www.webservicex.net/globalweather.asmx 上的 WebService 可提供天气预报服务。譬如，通过 Constant.java 中的 public class Constant {} ,设置 SOAP Action 要调用的方法名、命名空间以及 Web Service URL 值，其源代码为：

```
public static final String SOAP_ACTION=
"http://www.webserviceX.NET/GetCitiesByCountry";
public static final String METHOD_NAME=
"GetCitiesByCountry";
public static final String SOAP_ACTION2=
"http://www.webserviceX.NET/GetWeather";
public static final String METHOD_NAME2="GetWeather";
public static final String NAMESPACE=
"http://www.webserviceX.NET";
public static final String URL=
"http://www.webservicex.net/globalweather.asmx";
```

然后在 WebServiceCaller.java 中实例化一个 SoapSerializationEnvelope 对象，设置 SoapObject 的命名空间、方法名、参数等；并通过实例化一个 AndroidHttpTransport 对象来调用 Web Service，并获得 xml 字符串数据，其代码段如下：

```
AndroidHttpTransport androidHttpTransport=
new AndroidHttpTransport (Constant.URL);
try {
androidHttpTransport.call(soapAction, envelope);
Object result=envelope.getResponse();
xmlStr=result.toString();
} catch(Exception e) {
e.printStackTrace();
}
```

4.4 解析数据列表

通过调用天气预报 Web Service 服务查询到北京市天气预报的 Xml 数据结构如下：

```
<string><?xml version="1.0" encoding="utf-16"?>
<CurrentWeather>
  <Location >Beijing, China (ZBAA) 39 -56N 116 -17E
55M</Location>
  <Time>Jan 20, 2013 - 08:00 PM EST / 2013.01.21 0100
UTC</Time>
  <Wind> from the NNW (330 degrees) at 9 MPH (8 KT):
0</Wind>
  <Visibility> 1 mile(s):0</Visibility>
  <Temperature> 26 F (-3 C)</Temperature>
  <Wind>Windchill: 17 F (-8 C):1</Wind>
  <DewPoint> 23 F (-5 C)</DewPoint>
  <RelativeHumidity> 86%</RelativeHumidity>
  <Pressure> 30.39 in. Hg (1029 hPa)</Pressure>
  <Status>Success</Status>
```

</CurrentWeather></string>

以上是典型的以 Xml 文档形式返回的数据。Android 操作系统对 xml 字符串数据的操作功能很强，提供了 dom、sax 以及 xmlpul 3 种方式。Android SDK 提供了 android.sax 包以方便 SAX Handler 的开发，可用来解析 xml 结果字符串。在本设计实例的 CitiesWeatherForecast.java 文件中，解析 xml 数据时，先通过 RootElement root=new RootElement（" NewDataSet"）获得 xml 数据的根节点；然后再寻找其子节点直到找到 City 子节点，并设置一个监听器 setEndTextElementListener（）来获得 City 子节点的值；最后使用 org.xml.sax 的 SAXParser 来解析 xml 数据，把数据存入 List<String>并返回：

```
SAXParserFactory factory=SAXParserFactory.newInstance();
  SAXParser parser=factory.newSAXParser();
  XMLReader xmlreader=parser.getXMLReader();
  xmlreader.setContentHandler(root.getContentHandler());
  InputSource is=new InputSource
  (new StringBufferInputStream(xmlStr));
  xmlreader.parse(is);
```

4.5 在 UI 上呈现城市列表数据

当获得 List<String>类型的数据之后，就可以把它绑定到 main.xml 定义的 ListView 组件上，然后使用 ListActivity 呈现列表数据。在 WeatherForecast.java 中的代码段为：

```
List<String> cityList=getAllCitiesNames();
setListAdapter(new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, cityList));
```

即使用方法 setListAdapter（）把一个 ListActivity 填充进去。

4.6 城市天气实况预报系统的调试

城市天气实况预报系统的测试结果如下。在 Eclipse 的 Package Explorer 窗口中用鼠标右键选择 CitiesWeatherForecast 工程名，在弹出的窗口中选择“Run As”→“Android Application”安装该应用程序到 Android 模拟器并启动它。然后单击所要查询的城市名称列表项，稍等片刻便会显示出该城市的天气实况信息。也可以在文本框中直接输入所要查询的城市名称，单击“Search”同样会获得相应城市的天气预报信息。

5 结语

Android 平台具有极大的开放性和兼容性，并且本身大量使用了开源代码库，深为开发人员所喜爱。Android 智能手机应用程序的开发涉及了它的整个体系结构，是一项非常复杂的工程。在介绍基于 Android 平台的应用程序设计原理的基础上，提出了 Android 用户界面设计、获取并解析城市列表数据的一种方法，给出了在用户界面上呈现数据的原理与设计过程，最后通过模拟器进行了应用程序的调试。当然，基于 Android 平台的开发技术还需要进一步完善，需要在日后的工作中不断探索、研究，以建立实用的城市天气实况预报系统。

参考文献

[1] 杨丰盛. Android 应用开发揭秘. 机械工业出版社.

[2] W.Frank Ableson Charlie Collins Robi Sen . Google Android 揭秘. 人民邮电出版社.

[3] Burnette Ed . 张波，高朝勤，杨选，等，译. Android 基础教程. 人民邮电出版社，2009.

[4] 张利国，代闻，龚海平，编著. Android 移动开发案例详解. 人民邮电出版社，2010.

[5] 向辉，张国印，陈洁. Android 应用程序开发. 清华大学出版社，2010.

（收稿日期：2013-01-25）

（上接第 59 页）

设置等由于比较简单不在此进行介绍。

4 结语

该模型主要借鉴了 I/O 完成端口的思想，通过将协议性串口通信的每次通信过程抽象成通信元提交给通信元队列进行自动处理，从而实现对底层通信细节的封装，实现串口的模块化。将该模块应用于工业造气炉监控组态软件设计中，实际运行表明：该模型具有简单的接口和稳定、高效的运行机制。

参考文献

[1] Jeffrey Richter. Advanced Windows (3rd edition) [M] . USA: Microsoft Press, 1997.

[2] Jim Beveridge, Robert Wiener. Multithreading Applications in Win32: The Complete Guide to Threads (Addison-Wesley Microsoft Technology Series) [M] . USA: Addson-Wesley Professional, 1996.

[3] Anthony Jones, Jim Ohlund. Network Programming for Microsoft Windows (2nd edition) [M] . USA: Microsoft Press, 2002.

[4] Anthony Jones, Amol Deshpande. Windows Sockets 2.0: Write Scalable Winsock Apps Using Completion Ports [J] . MSDN Magazine, Microsoft Press, October 2000: 30-35.

[5] Ruediger Asche. Writing Windows NT Server Applications in MFC Using I/O Completion Ports [J] . MSDN Magazine, Microsoft Press, September, 1996: 25-30.

基金项目：厦门市科技计划项目（3502Z20103002）资助。

（收稿日期：2013-01-04）

来卡网出品 LAICAR.COM shop35833438.taobao.com

基于 WAMP 的在线评测系统设计与实现

李臣龙　鲍广喜

摘　要：详细阐述了在线评测系统的结构原理，运用 PHP、MySQL 和 Apache 组合设计实现了基于 B/S 结构的在线评测系统，实现了用户在线评测、查看状态、讨论等功能。

关键词：ACM/ICPC 竞赛；Online Judge 系统；PHP 语言；MySQL 数据库；Apache 软件

1　引言

在线评测系统源于 ACM/ICPC（国际大学生程序设计竞赛）是由 ACM (Association for Computing Machinery，美国计算机协会) 组织的年度性竞赛，始于 1970 年，是全球大学生计算机程序能力竞赛活动中最有影响的一项赛事。

在线评测系统能够实时对用户提交的程序进行评测，对提高同学编程能力有很大帮助，同时也解决手工评测效率的问题。本设计旨在提供一个训练的平台，也为希望提高计算机程序设计能力的同学提供一个练习的场所。

2　开发工具

“针对 ACM/ICPC 的在线评测系统”是基于 Internet 的 B/S 模式的。客户端可以直接在浏览器中粘贴相关题目的源代码，服务器接受到源代码之后将记录存入数据库，然后编译，执行，并验证源代码的正确性。服务器端采用的是 Apache+PHP+MySQL 的组合方案。下面对用到的相关工具做简要介绍。

2.1　PHP 语言

本系统采用跨平台的服务器端嵌入式语言 PHP (Hypertext Preprocessor) 作为主开发语言。这是因为尽管目前的网络数据库开发工具众多 (例如：应用较广泛的有 ASP、PHP、JSP、CGI 等)，但 PHP 秉承了 Linux 的 GNU 风格，借助于公开的源代码，大量应用 C,C++Java 的语法，具有易学、好用的优点。PHP 内置了文件上传密码认证、Cookie 操作、动态 GIF 生成、邮件收发、XML，共享内存等功能，而更为重要的是 PHP 可以直接连接多种数据库，并完全支持 ODBC,PHP 目前所支持的数据库有 Or2cale，MySQL，Sybase，Informase/ Illustra 等。

2.2　MySQL 与 Apache

本系统采用 MySQL 作为 SQL 数据库服务软件。这是因为 MySQL 具有多用户、多线程、速度快、易用性强的优点。Apache 虽与微软公司的 IIS 类似，同属于一种 Web 服务器，但 Apache 的优点使其成为已为目前最流行的 Web 服务器，故本系统采用 Apache 作为 Web 服务器[2]。

3　系统原理

本系统采用流行的 B/S 模式。只要在客户端通过 IE 浏览器便可以访问和管理，大量的操作放在服务器端，包括请求反馈、数据存取、评判结果生成等等，只是将用户需要的结果通过 HTML 文档页面显示在客户端。基本工作模式如图 1 所示。

学生通过浏览器向 Web 服务器发出请求，服务器读取相应 PHP 文件，使用脚本解释引擎 PHP 文件进行解释执行，对后台数据库执行 SQL 命令，调用符合用户的应用服务，然后将结果返回浏览器。B/S 结构充分利用了服务器资源，实现了开发环境与应用环境的分离，易于维护，信息共享度高。

管理员可以通过浏览器可以对用户信息、题库等进行管理，负责整个系统的数据操作和维护，这样安全性更高，能够实现较为复杂管理机制。



图 1　系统工作模式

4　系统主要功能设计与实现

本设计主体思想是为了方便用户，提供一个练习、交流并提高编程水平的平台。设置有用户注册、题目浏览、提交、查看提交状态、查看排名、论坛等功能。

4.1　注册、登录与提交模块

4.1.1 注册模块

该模块的功能是实现考生注册。学生输入想要注册的用户名和密码等信息，提交之后由系统调用服务器 MySQL 数据库判断该用户是否已经存在，如果不存在，就初始化 Session 环境变量，注册 session 变量，同时把用户输入的用户名和密码写到 MySQL 数据库中。

4.1.2 登录模块

登录模块采用多级用户权限管理，从管理效率，数据安全上考虑，本系统采用严格的权限分级制度，不同用户对应不同

LAICAR.COM
shop35833438.taobao.com

活动权限，以确保数据安全。在 B/S 模式下根据用户 ID 赋予用户不同权限，例如学生登录后可以浏览题目、查看自己状态、提交题目、参加 BBS 讨论等。管理员可以对数据库和系统资源进行管理，包括学生权限和资料管理，题目更新管理，消息和平台内公共资源管理等。

根据登陆信息判断其合法性,然后在数据库中查找与之相匹配的记录，进行身份判断,定义用户权限级别及初始化 Session 环境变量等。如果没有该用户则提示其进行注册。判断用户类型 (学生、管理员) 、根据用户类型重定向用户首页(学生首页、管理员首页)。

4.1.3 提交模块

用户注册登录后可以浏览题目进行程序提交，提交信息存入个人数据库表中，程序交给后台评测系统进行处理。

4.2 后台评测系统

服务器后台运行守候程序，每一个时间间隔查询一次数据库，如发现有提交纪录，这启动评判进程，进行相关题目的评判。每道题目均对应一个数据输入输出文件，让评判进程启动一个子进程编译运行用户提交上来的程序，当程序启动时计时，在时间允许范围之内，如果其得到了正确答案，则修改数据库，令其 status 为 Accepted，否则，答案错，或者表达错，运行错等流程图（如图 2 所示）。

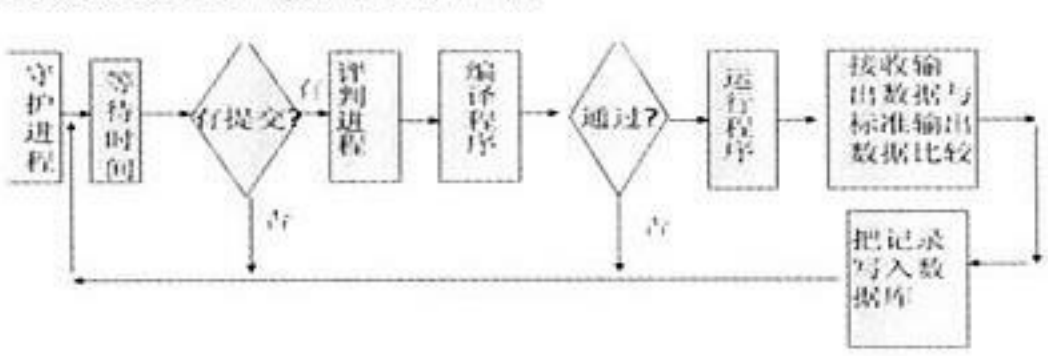


图 2 后台守候程序运行流程

4.3 关键技术

4.3.1 程序编译

本设计实现了编译运行 C、C++、Java 程序的在线编译，下面以对 C/C++程序为例说明 Web 编译的实现。系统使用 VC 编译器，通过命令 cl 对代码进行编译运行，最后 PHP 调用外部命令 passthru 显示编译信息。

对提交程序转换为用户选择的语言格式保存到数据库，然后通过 PHP 调用 VC 的 cl 命令进行编译与运行：

```
<?
 include("topno.php");  include("runtime.php");
 global $language, $source;
if(strlen($source)<1)
{
表单提交代码等信息
}
else{
if($language==4)// 针对某种语言产生编译文件
 {
 $file=fopen("data/$problem_id/$problem_id.cpp","w");
 fwrite($file,$source);
 fclose($file);
  }
  …//其他语言
}
chdir("data/$problem_id");
$command = "my1.bat $problem_id.cpp";
exec($command);//执行外部命令编译程序
global $file;
$dir=dir(".");
while($file=$dir->read())//读取服务器端多个验证文件进行
//验证
{
if(strstr($file,".in"))
{
$command="$problem_id.exe <$file >datatemp.out";
$runtime = new Runtime(3);
passthru($command); 执行外部命令运行程序
echo $runtime->totaltime();
$dout=$file;  $dout=rtrim($dout,".in");
$dout=$dout.".out";
global $text1;global $text2;global $size1;global $size2;global
$dtemp;global $dsamp;
$arrtext1 = file("datatemp.out");  $arrtext2 = file("$dout");
for ($i=0; $i < count($arrtext2); $i++)  //通过逐字符比较判断
//答案
{   $trimed1=rtrim($arrtext1[$i],"\n");//消除空格回车符的影响
    $trimed1=rtrim($trimed1,"\r");
    $trimed1=rtrim($trimed1," ");
    if($i! =(count($arrtext1)-1))
    $trimed1=$trimed1."\r"."\n";
    $trimed2=rtrim($arrtext2[$i],"\n");
    $trimed2=rtrim($trimed2,"\r");
    $trimed2=rtrim($trimed2," ");
    if($i! =(count($arrtext2)-1))
    $trimed2=$trimed2."\r"."\n";
     if($trimed1! =$trimed2)
     {
          成功
     }
     else  continue;
     }
} }}
其他处理……
?>
```

4.3.2 程序评判

本设计采用逐字符比较的方式对程序进行评判。即评判程序每次从管道读入一个字符与标准输出数据比较，如果两个文

（下转第 83 页）

来卡网出品
LAICAR.COM
shop35833438.taobao.com

GIF 与 BMP 互转程序开发

江 洪

摘 要：结合 GIF 与 BMP 文件的结构，使用 VC6.0 开发了一个简单的互转程序。程序支持 256 色 GIF 与 BMP 之间的相互转换。通过程序可以了解 GIF 与 BMP 文件的结构，以及相应的 LZW 压缩和解压方法，对更高层次的程序开发有一定的参考价值。

关键词：GIF 文件；BMP 文件；LZW 算法；转换

1 引言

GIF 文件是 CompuServe 公司开发的图形文件格式。该文件中包含调色板，最多支持 256 色。文件内部分成许多块，并按一定的规则存储，图像数据采用 LZW 算法进行压缩存放。LZW 算法是一种无损压缩算法，它使用了一个动态的字典，记录文件中出现的字符串，并将再次出现的字符串用字典中的索引替代，从而减小文件大小。

BMP 文件是 MicroSoft 公司开发的图形文件格式。这是一种基本的图形文件格式。BMP 图像数据一般不进行压缩，直接用 RGB 颜色分量或调色板索引表示每个像素。

下面开发一个实现 256 色 GIF 与 BMP 文件相互转换的程序。

2 GIF 文件结构

GIF 文件由若干数据块组成，文件结构如表 1 所示。

表 1 GIF 文件结构

	字段	长度	说明
文件头	signature	3 字节	GIF 标识，值固定为 GIF
	version	3 字节	版本号，值为 87a 或 89a
逻辑屏幕标识符	width	2 字节	定义屏幕宽度
	height	2 字节	定义屏幕高度
	m	1 字节	第 7 位为全局调色板标志，置位表示使用全局调色板；
	cr		第 4、5、6 位表示颜色深度，cr+1 确定颜色位数
	s		第 3 位为分类标志，置位表示调色板分类排列
	pixel		第 0、1、2 位表示全局调色板大小，调色板索引数为 $2^{pixel+1}$
	backcolor	1 字节	背景色索引
	ratio	1 字节	像素宽高比
全局调色板	r	1 字节	红色分量
	g	1 字节	绿色分量
	b	1 字节	蓝色分量
		...	共有 256*3=768 个字节的调色板
图形控制扩展块	id	1 字节	扩展块标识，固定值 0x21，需要 89a 版本支持
	label	1 字节	扩展标签，固定值 0xf9
	blocksize	1 字节	块大小，固定值 4
	reserved	1 字节	第 5、6、7 位保留，置 0
	dealmethod		第 2、3、4 位表示处置方法
			0-不处置 1-不处置图形，把图形从当前位置移去
			2-回复到背景色 3-回复到先前状态
	i		第 1 位表示用户输入标志，置位表示期待
	t		第 0 位表示透明色标志，置位表示使用透明色
	delaytime	2 字节	每帧延迟时间，以 0.01 秒为单位
	transcolor	1 字节	透明色索引
	terminator	1 字节	块终结器，固定值 0
图像标识符	id	1 字节	图像标识符，固定值 0x2c
	x	2 字节	X 方向偏移量，设 0
	y	2 字节	Y 方向偏移量，设 0
	width	2 字节	图像宽度
	height	2 字节	图像高度
	m	1 字节	第 7 位为局部调色板标志，置位表示使用局部调色板，仅供本帧图像使用，为 0 使用全局调色板
	i		第 6 位为交织标志，置位图像数据使用交织排列，否则使用顺序排列
	s		第 5 位为分类标志，置位表示调色板分类排列
	reserved		第 3、4 位保留，设 0
	pixel		第 0、1、2 位表示局部调色板大小，调色板索引数为 $2^{pixel+1}$
局部调色板	r	1 字节	红色分量
	g	1 字节	绿色分量
	b	1 字节	蓝色分量
		...	共有 256*3=768 个字节的调色板

shop35833438.taobao.com

	字段	长度	说明
图像数据块		若干字节	图像数据使用 lzw 算法进行压缩后存放，具体存放方法随后详细讲述
		...	如 GIF 有多帧图像，则可以有多个图形控制扩展块、多个图像标识符、多个局部调色板、多个图像数据块
文件尾	tail	1 字节	GIF 文件结尾，固定值 0x3b

图像数据块结构：

第 1 个字节表示颜色位数，如 256 色该值就为 8。

然后就是若干个数据包，每个数据包不超过 255 个字节。每个数据包第 1 个字节表示包长度，包长度之后为包数据，数据是经过 LZW 压缩的数据流。

最后 1 个字节为图像数据块结束符，固定值 0。

3 BMP 文件结构

如表 2 所示。

表 2 BMP 文件结构

	字段	大小	说明
文件头	id	2 字节	BMP 文件标识，值为 'BM'
	filesize	4 字节	整个 BMP 文件的大小
	reserved	4 字节	保留，值为 0
	dataoffset	4 字节	表示位图数据距离文件头的字节数
信息头	headsize	4 字节	信息头长度，值设为 40
	width	4 字节	位图宽度，以像素为单位
	height	4 字节	位图高度，以像素为单位
	planes	2 字节	位平面数，值设为 1
	bits	2 字节	每个像素占用的位数
	compress	4 字节	压缩方式 设 0
	datasize	4 字节	位图数据大小
	hresolution	4 字节	水平分辨率
	vresolution	4 字节	垂直分辨率
	colorused	4 字节	使用的颜色数
	colorimportant	4 字节	重要的颜色数
调色板	rgbblue	1 字节	蓝色分量
	rgbgreen	1 字节	绿色分量
	rgbred	1 字节	红色分量
	rgbreserved	1 字节	保留，值为 0
图像数据		若干字节	每行数据字节数应为 4 的整数倍，如果不是 4 的倍数则用 0 进行填充

4 LZW 算法

GIF 文件中图像数据采用 LZW 算法进行压缩存储。这是一种无损的数据压缩算法。也就是经过解压后的数据和原始数据完全一样。该算法使用了一个动态字典，字典中存储了文件中曾经出现的字符串信息。这样，该字符串再次出现时，就可以用字典中的标号代替，这就达到了数据压缩的目的。但这个字典并不单独存储在文件中，而是在数据压缩和解压的过程中动态生成。

LZW 算法中，编码不是按字节存放的，而是按位存放的。存放的顺序是从右向左，每过 8 位再从下一字节从右向左存放。而且，编码的位数是可变长度的。编码的位数由字典中的总项数决定。

字典用如下结构体表示：

```
struct dictionarystru  //字典结构
{
  short prefix;  //前缀
  short suffix;  //后缀
  short len;  //串长度
  unsigned char *str;  //串内容
};
```

该字典是总项数为 4096 的一个结构数组。

前 256 项代表单个字符，称为根编码。

第 257 项称为字典复位标志，当字典总项数超过 4096 时输出此标志，之后以前定义的字典作废，从头开始建立字典。

第 258 项称为图像结束标志，输出该标志后 GIF 图像就结束了。

从 259 项开始，直到 4096 项，每一项表示数据中曾经出现的一个字符串，每个字符串至少为 2 个字符。

前缀为每个字符串的前趋字符串编码，可以代表多个字符。后缀为每个字符串的后继字符串编码，应为单个字符。

编码位数根据字典中总项数动态变化。258<=字典总项数<=512，编码位数设 9；512<字典总项数<=1024，编码位数设 10；1024<字典总项数<=2048，编码位数设 11；2048<字典总项数<=4096，编码位数设 12。字典总项数>4096，字典就复位了，这时编码位数又重新变回 9。编码位数就这样不断变化，直到图像数据结束。

4.1 LZW 编码的流程

（1） 初始化字典，编码位数设为 9，前缀设为-1。

（2） 检查字典总项数。258<=字典总项数<=512，编码位数设 9；512<字典总项数<=1024，编码位数设 10；1024<字典总项数<=2048，编码位数设 11；2048<字典总项数<=4096，编码位数设 12。

（3）读一个编码，如为第一个编码，将该字符赋给后缀，在字典中查找，该编码为根编码，在字典中已存在，将前缀变为在字典中的序号，输出第一个字典复位标志。如不为第一个编码，将该字符赋给后缀。字典已满，输出字典复位标志。在字典中查找。字典中找不到，将前缀+后缀表示的串加入字典，输出前缀，后缀变前缀；字典中能找到，把字典中序号赋给前缀。如为最后一个字符，输出图像结束标志，跳到步骤 4，否则跳到步骤 2。

来卡网出品 LAICAR.COM shop35833438.taobao.com

(4) 结束编码。

4.2 LZW 解码流程

(1) 读第一个编码，第一个编码应为字典复位标志，初始化字典，编码位数变为 9，输出下一个编码，将下一个编码赋给前缀。

(2) 读下一个编码。读到字典复位标志，初始化字典，编码位数变为 9，输出下一个编码，将下一个编码赋给前缀。如该编码为根编码，将该字符赋给后缀，将前缀+后缀表示的串加入字典，将该编码赋给前缀，直接输出该编码；如该编码不为根编码，则判断如果编码为 258，则将前缀作为新串的前缀，字典中该编码项前缀作为新串的后缀，加入字典中，否则将前缀作为新串的前缀，字典中前一编码项后缀作为新串的后缀，加入字典中，输出该编码所表示的串。如读到图像结束标志，跳到步骤 4。

(3) 设置编码位数。258<=字典总项数<512，编码位数设 9；512<=字典总项数<1024，编码位数设 10；1024<=字典总项数<2048，编码位数设 11；2048<=字典总项数<4096，编码位数设 12，跳到步骤 2。

(4) 结束解码

下面结合一个实际的例子，加深一下对 LZW 算法的理解。

比如编码前数据如下：

54 54 54 54 54 54 54 54 54 0 54 54 54 54 54 178 54 178 0 54 178 178 54 0

第 1 步 读入 54，prefix=-1，suffix=54，在字典中能找到，于是输出 256，prefix=54。

第 2 步 读入 54，prefix=54，suffix=54，在字典中找不到，于是将 (54,54) 加入字典 258 项，输出 54，prefix=54。

第 3 步 读入 54，prefix=54，suffix=54，在字典中找到 258 项，于是 prefix=258。

第 4 步 读入 54，prefix=258，suffix=54，在字典中找不到，于是将 (258,54) 加入字典 259 项，输出 258，prefix=54。

第 5 步 读入 54，prefix=54，suffix=54，在字典中找到 258 项，于是 prefix=258。

第 6 步 读入 54，prefix=258，suffix=54，在字典中找到 259 项，于是 prefix=259。

第 7 步 读入 54，prefix=259，suffix=54，在字典中找不到，于是将 (259,54) 加入字典 260 项，输出 259，prefix=54。

第 8 步 读入 54，prefix=54，suffix=54，在字典中找到 258 项，于是 prefix=258。

第 9 步 读入 54，prefix=258，suffix=54，在字典中找到 259 项，于是 prefix=259。

第 10 步 读入 0，prefix=259，suffix=0，在字典中找不到，于是将 (259,0) 加入字典 261 项，输出 259，prefix=0。

第 11 步 读入 54，prefix=0，suffix=54，在字典中找不到，于是将 (0,54) 加入字典 262 项，输出 0，prefix=54。

第 12 步 读入 54，prefix=54，suffix=54，在字典中找到 258 项，于是 prefix=258。

第 13 步 读入 54，prefix=258，suffix=54，在字典中找到 259 项，于是 prefix=259。

第 14 步 读入 54，prefix=259，suffix=54，在字典中找到 260 项，于是 prefix=260。

第 15 步 读入 54，prefix=260，suffix=54，在字典中找不到，于是将 (260,54) 加入字典 263 项，输出 260，prefix=54。

第 16 步 读入 178，prefix=54，suffix=178，在字典中找不到，于是将 (54,178) 加入字典 264 项，输出 54，prefix=178。

第 17 步 读入 54，prefix=178，suffix=54，在字典中找不到，于是将 (178,54) 加入字典 265 项，输出 178，prefix=54。

第 18 步 读入 178，prefix=54，suffix=178，在字典中找到 264 项，于是 prefix=264。

第 19 步 读入 0，prefix=264，suffix=0，在字典中找不到，于是将 (264,0) 加入字典 266 项，输出 264，prefix=0。

第 20 步 读入 54，prefix=0，suffix=54，在字典中找到 262 项，于是 prefix=262。

第 21 步 读入 178，prefix=262，suffix=178，在字典中找不到，于是将 (262,178) 加入字典 267 项，输出 262，prefix=178。

第 22 步 读入 178，prefix=178，suffix=178，在字典中找不到，于是将 (178,178) 存入字典 268 项，输出 178，prefix=178。

第 23 步 读入 54，prefix=178，suffix=54，在字典中找到 265 项，于是 prefix=265。

第 24 步 读入 0，prefix=265，suffix=0，在字典中找不到，于是将 (265,0) 存入字典 269 项，输出 265，prefix=0。

执行完第 24 步后，输出应为 256 54 258 259 259 0 260 54 178 264 262 178 265，这个结果还要以 9 位一个编码的规则进行重新排列，因此重新排列后输出变为 0 109 8 28 56 16 0 65 27 178 16 26 148 149 16。

解码是编码的逆过程。

0 109 8 28 56 16 0 65 27 178 16 26 148 149 16 经过 9 位一个编码重新排列后变为 256 54 258 259 259 0 260 54 178 264 262 178 265。

第 1 步 读入 256，初始化字典，输出下一个编码 54，prefix=54。

第 2 步 读入 258，非根编码，于是将 (54,54) 存入字典 258 项，prefix=258，输出 54 54。

第 3 步 读入 259，非根编码，于是将 (258,54) 存入字典 259 项，prefix=259，输出 54 54 54。

第 4 步 读入 259，非根编码，于是将 (259,54) 存入字典 260 项，prefix=259，输出 54 54 54。

第 5 步 读入 0，根编码，于是将 (259,0) 存入字典 261

项，prefix=0，输出 0。

第 6 步 读入 260，非根编码，于是将 (0,54) 存入字典 262 项，prefix=260，输出 54 54 54 54。

第 7 步 读入 54，根编码，于是将 (260,54) 存入字典 263 项，prefix=54，输出 54。

第 8 步 读入 178，根编码，于是将 (54,178) 存入字典 264 项，prefix=178，输出 178。

第 9 步 读入 264，非根编码，于是将 (178,54) 存入字典 265 项，prefix=264，输出 54 178。

第 10 步 读入 262，非根编码，于是将 (264,0) 存入字典 266 项，prefix=262，输出 0 54。

第 11 步 读入 178，根编码，于是将 (262,178) 存入字典 267 项，prefix=178，输出 178。

第 12 步 读入 265，非根编码，于是将 (178,178) 存入字典 268 项，prefix=265，输出 178 54。

执行完第 12 步后，输出应为 54 54 54 54 54 54 54 54 54 0 54 54 54 54 54 178 54 178 0 54 178 178 54，与编码前数据完全一致。

5 关键代码

限于篇幅，只列出 LZW 编码和解码部分代码，其他代码可参见源程序（略）。

LZW 编码函数如下：

```
void gifencod e (unsigned char *prevbuf,unsigned long prevbuflen,
     unsigned char *postbuf,unsigned long *postbuflen)
{
 unsigned long i,len,pos,pos1,k;
 short prefix,suffix,n,c,bits,j,count,len1;
 unsigned char *b,ch;
 b=(unsigned char *)malloc(50000000);
 for(i=1;i<=50000000;i++) b[i-1]=0;
 len=0;postbuf[len]=8;len++;
 initdictionary(); //初始化字典
 pos=0;pos1=0;prefix=-1;
 for(i=1;i<=prevbuflen;i++) //顺序读取 prevbuf
 {
  //设置编码位数
if(entrycount>=258&&entrycount<=512) bits=9;
  if(entrycount>512&&entrycount<=1024) bits=10;
  if(entrycount>1024&&entrycount<=2048) bits=11;
  if(entrycount>2048&&entrycount<=4096) bits=12;
  suffix=(short)(prevbuf[i-1]&0xff);
  n=finddictionary(prefix,suffix); //查字典
  if(n==-1)
  {
   //字典中无该项,将该项加入字典,输出前缀
   dictionary[entrycount].prefix=prefix;
   dictionary[entrycount].suffix=suffix;
   entrycount++;
   c=prefix;
   for(j=1;j<=bits;j++)
   {
  b[pos]=(unsigned char)((c>>(j-1))&0x1);pos++;
   }
   prefix=suffix; //后缀变前缀
  }
  //字典中有该项,前缀变为该项在字典中序号
  if(n>=0&&n<=4095) prefix=n;
 if(entrycount>4096)
  {
   //字典已满,输出字典复位标志
   initdictionary();
   c=256;
   for(j=1;j<=bits;j++)
   {
  b[pos]=(unsigned char)((c>>(j-1))&0x1);pos++;
   }
  }
  if(i==1) //开始先输出一个字典复位标志
  {
   c=256;
   for(j=1;j<=bits;j++)
   {
  b[pos]=(unsigned char)((c>>(j-1))&0x1);pos++;
   }
  }
  if(pos-pos1>=2040||i==prevbuflen)
  {
   postbuf[len]=(unsigned char)255; //输出包长度
   len1=255;
   if(i==prevbuflen) //读到最后一个字符
   {
      //输出前缀
      c=prefix;
    for(j=1;j<=bits;j++)
      {
   b[pos]=(unsigned char)((c>>(j-1))&0x1);pos++;
      }
      //输出图像结束标志
  c=257;
    for(j=1;j<=bits;j++)
      {
   b[pos]=(unsigned char)((c>>(j-1))&0x1);pos++;
      }
  len1=(short)(((pos-pos1)/8)&0xff);
      if((pos-pos1)%8! =0) len1++;
      postbuf[len]=(unsigned char)len1; //输出包长度
  }
  len++;count=0;k=pos1;
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
 while(1)  //循环输出编码后字符
 {
ch=0;
      ch=ch+(b[k]<<0);ch=ch+(b[k+1]<<1);
      ch=ch+(b[k+2]<<2);ch=ch+(b[k+3]<<3);
      ch=ch+(b[k+4]<<4);ch=ch+(b[k+5]<<5);
      ch=ch+(b[k+6]<<6);ch=ch+(b[k+7]<<7);
      postbuf[len]=ch;len++;count++;k=k+8;
      if(count>=len1) break;
       }
       pos1=pos1+2040;
      }
 }
 postbuf[len]=0;len++;  //输出块结束符
 *postbuflen=len;  //设置编码后缓冲区长度
 free(b);
}
```

LZW 解码函数如下：

```
void  gifdecod e  (unsigned  char  *prevbuf,unsigned  long
prevbuflen,
      unsigned char *postbuf,unsigned long *postbuflen)
{
 unsigned long i,pos,pos1,len;
 unsigned short bits,c,c1,prefix,j;
 unsigned char *b,ch;

b=(unsigned char *)malloc(50000000);
pos=0;
//将解码前缓冲区按位存入缓冲区 b
for(i=1;i<=prevbuflen;i++)
{
    ch=prevbuf[i-1];
    for(j=1;j<=8;j++) b[pos+j-1]=(ch>>(j-1))&1;
    pos=pos+8;
}
pos=0;len=0;bits=9;
while(1)  //循环解码
{
    //读一个编码
    c=0;
    for(j=1;j<=bits;j++) c=c+(b[pos+j-1]<<(j-1));
  if(c==257) break;  //读到图像结束标志退出
if(c>=0&&c<4096)  //0<=编码<4096
  {
   if(c>=0&&c<=255)
   {
       /* 0<=编码<=255,将 prefix 作为前缀,c 作为后缀
       加入字典,前缀变为 c */
     dictionary[entrycount].prefix=prefix;
     dictionary[entrycount].suffix=(unsigned char)(c&0xff);
   adddictionarystr();
     entrycount++;prefix=c;
      //0<=编码<=255 直接输出
postbuf[len]=(unsigned char)(c&0xff);len++;
 }
 if(c>=258&&c<4096)  //258<=编码<4096
 {
  /* 将 prefix 作为前缀,
如当前项为第 258 项,将字典 c 项的 prefix 作为后缀
      否则将字典 c-1 项的 suffix 作为后缀,加入字典 */
   dictionary[entrycount].prefix=prefix;
    if(c>258)
     dictionary[entrycount].suffix=
     (unsigned char)(dictionary[c-1].suffix&0xff);
    else
     dictionary[entrycount].suffix=
     (unsigned char)(dictionary[c].prefix&0xff);
adddictionarystr();
    entrycount++;prefix=c;
    //255<=编码<4096 输出该编码所表示的串
  for(j=1;j<=dictionary[c].len;j++)
    {
     postbuf[len]=
     (unsigned char)(dictionary[c].str[j-1]&0xff);
     len++;
    }
 }
}
if(c==256)  //读到字典复位标志
  {
   pos=pos+bits;pos1=pos;
   initdictionary();  //初始化字典
   bits=9;
   //输出下一个编码
   c1=0;
   for(j=1;j<=bits;j++) c1=c1+(b[pos1+j-1]<<(j-1));
   postbuf[len]=(unsigned char)(c1&0xff);
   len++;prefix=c1;
  }
pos=pos+bits;
  //设置编码位数
if(entrycount>=258&&entrycount<512) bits=9;
  if(entrycount>=512&&entrycount<1024) bits=10;
  if(entrycount>=1024&&entrycount<2048) bits=11;
    if(entrycount>=2048&&entrycount<4096) bits=12;
 }
 *postbuflen=len;  //设置解码缓冲区长度
}
```

6 结语

程序实现了既定的功能。希望本例程对读者有所帮助。

（收稿日期：2013-02-21）

LAICAR.COM
shop35833438.taobao.com

Vega Prime 之 SpeedTree 模块程序设计

赵常寿 张 鹏 吴红权

摘 要：SpeedTree 模块能够在实时帧率下进行真实感植被景观的定义与渲染，生成具备碰撞映射、阴影和精细纹理的植被效果。结合实例，介绍了 Vega Prime 中 SpeedTree 模块的基本程序设计方法，在此基础上可以开发 SpeedTree 模块的具体应用。

关键词：Vega Prime 平台；SpeedTree 模块；模型定义文件；交叉引用文件

1 引言

SpeedTree 是美国 IDV（Interactive Data Visualization）公司研发制作的一款专门的三维树木建模软件，支持大片树木的快速建立和渲染，而且它本身还带有强大的树木库。不仅可以通过插件将树木导入到其他的三维建模软件中使用，也可以为游戏引擎提供强大的树库支持，目前已经成为著名游戏引擎 Unreal 的御用树木生成软件。

Vega Prime 是 Presagis 公司研发制作的一款用于创建及配置视景仿真、多领域仿真和通用可视化应用的视景平台，利用其跨平台性与可扩展性，Vega Prime 成为实时 3D 应用开发与调度最佳的 COTS 类工具包。Vega Prime 开发环境大大提高进行实时 3D 应用的创建与配置速度，同时提供功能强大的扩展模块，以解决实时 3D 应用开发中遇到的难题，其中的 SpeedTree 模块能够在实时帧率下进行真实感植被景观的定义与渲染。SpeedTree 模块集成来自 IDV 公司的获奖产品 SpeedTree 技术，此技术目前已经成为 US DoD 训练系统和大多数视景游戏的特定特征。SpeedTree 模块能对 Vega Prime 应用中高密度植被进行定义和渲染，能在达到最佳视觉效果的同时保持原有的渲染效率不变，能生成具备碰撞映射、阴影和精细纹理的植被效果，提供具有 200 种树和植物种类的模型库，包括阔叶树、针叶树、棕榈树、仙人掌和灌木，允许对现有树型进行修改并创建新的树型。SpeedTree 植被还能方便地添加到现有的 OpenFlight 和 MetaFlight 数据库中。

下面讨论 SpeedTree for Vega Prime 模块的程序设计方法。

2 SpeedTree 模块

要使用 SpeedTree for Vega Prime 模块，必须安装 Vega Prime 2.1.2 或以上版本。

2.1 SpeedTree 模型定义文件

SpeedTree 模型定义文件（SpeedTree Definition File）用来批量定义 spt 格式的源模型及其渲染属性，其文件名是 vsgn_speedtree_defs.txt。该文件名是不可更改的，OpenFlight 加载器在加载 SpeedTree 模型时，只查找文件vsgn_speedtree_defs.txt，否则，将会产生不可预测的结果。vsgn_speedtree_defs.txt 示例文件如下：

```
MODEL tree1.speedtree {
    SOURCE_FILE    Beech_RT.spt
    INSTANCE_CULLING    false
    TECHNIQUE_DETAIL    true
    TECHNIQUE_SELF_SHADOW    true
    TECHNIQUE_DETAIL    true
    SCALE    1.0
}
MODEL tree2.speedtree {
    SOURCE_FILE    RiverBirchHero_RT.spt
    INSTANCE_CULLING    false
    TECHNIQUE_NORMAL_MAP    true
    TECHNIQUE_SELF_SHADOW    true
    TECHNIQUE_DETAIL    true
    SCALE    1.0
}
MODEL tree3.speedtree {
    SOURCE_FILE    WillowOak_RT.spt
    INSTANCE_CULLING    false
    TECHNIQUE_NORMAL_MAP    true
    TECHNIQUE_SELF_SHADOW    true
    TECHNIQUE_DETAIL    true
    SCALE    1.0
}
MODEL tree4.speedtree {
    SOURCE_FILE    SugarPine_RT.spt
    INSTANCE_CULLING    false
    TECHNIQUE_NORMAL_MAP    true
    TECHNIQUE_SELF_SHADOW    true
    TECHNIQUE_DETAIL    true
    SCALE    1.0
}
```

其中，关键字 MODEL 定义一个新的模型实例，接着是标识模型的文件名，然后在“{”和“}”之间定义模型参数。

来卡网出品
LAICAR.COM
shop35833438.taobao.com

2.2 交叉引用文件

交叉引用文件（Cross Reference File）用来定义 OpenFlight 外部引用名字和替换文件名字之间的映射关系，OpenFlight 加载器使用 Vega Prime 提供的搜索路径尝试加载外部引用的替换文件。交叉引用文件的文件名是 vsgn_xref_map.txt，该文件名是不可更改的，OpenFlight 加载器在加载 SpeedTree 模型时，只查找文件 vsgn_xref_map.txt，否则，将会产生不可预测的结果。vsgn_xref_map.txt 示例文件如下所示：

```
tree01.flt tree1.speedtree
tree02.flt tree2.speedtree
tree03.flt tree3.speedtree
tree04.flt tree4.speedtree
```

其中，第一个字符串是 OpenFlight 模型数据库外部引用文件名，第二个字符串是替换文件的名字，用来加载代替前面的外部引用文件。当 OpenFlight 加载器读到外部引用节点时，它将用外部引用文件名同交叉引用文件中提供的文件名进行比较，如果找到匹配的项，就尝试用替换文件代替外部引用文件。

2.3 SpeedTreeControl 类

SpeedTreeControl 类用来控制 SpeedTree 模型的属性，该类必须存在且唯一，其定义的属性包括：Wind Effects、Normal Mapping、Detail Texture、Self-Shadow 和 Instance Culling。为了在 Vega Prime 的 Active Preview 里看到 SpeedTree 模型，在应用配置文件 ACF 里必须有一个 SpeedTreeControl 实例。

3 SpeedTree 模块程序

SpeedTree for Vega Prime 模块程序设计流程主要是：

(1) 用 SpeedTreeCAD 创建 spt 模型。

(2) 用 Creator 创建包含外部引用的 flt 模型。

(3) 编辑 vsgn_speedtree_defs.txt 和 vsgn_xref_map.txt 文件。

(4) 编辑 ACF 文件，添加在(2)中创建的 flt 模型，添加 SpeedTreeControl 实例。

(5) 在 Active Preview 里进行预览，或者用 VC 编程进行驱动。

3.1 创建 spt 模型

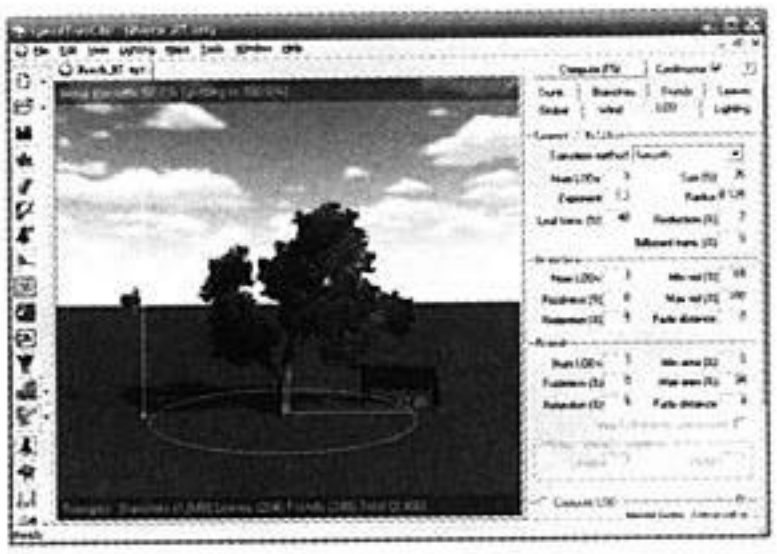


图 1 SpeedTreeCAD 4.1 运行界面

在 Vega Prime 2.2.1 版本中带了 SpeedTreeCAD 4.1 版本，可以用来创建 spt 模型，运行界面如图 1 所示。这不是讨论重点，所以不做详细讨论。

3.2 创建 flt 模型

flt 模型是 OpenFlight 格式的模型数据库，OpenFlight 格式是 Multigen Creator 建模软件独创的用于描述三维虚拟场景的层次化数据结构，已经成为事实上的业界标准数据格式。在 flt 模型数据库中可以添加对其他模型数据库的外部引用，外部引用的内容在层级视图中被放置在外部引用节点中，外部引用节点必须用跟组节点同级别的节点作为父节点。外部引用的模型对象在当前模型数据库中具有只读属性，不能在当前模型数据库中对其进行编辑，但是可以对外部引用进行整体的移动、旋转或者缩放。向模型数据库中添加外部引用的步骤如下：

(1) 在层级视图中选择适当的组节点，单击“Parent”按钮将其设为父节点。

(2) 单击菜单“File”－“External Reference”，打开外部引用对话框。

(3) 在外部引用对话框的“External File Name”编辑框中输入被引用的模型数据库文件名。

(4) 单击“OK”按钮完成外部引用。

还有一种常用的外部引用模型数据库的方法是直接在层级视图中，通过创建外部引用节点来进行外部引用操作。具体做法是，首先选择适当的组节点，然后打开创建工具箱，单击创建外部引用工具按钮。

下面具体创建一个包含外部引用的 flt 模型，使用的 Creator 版本为 3.2，运行界面如图 2 所示。

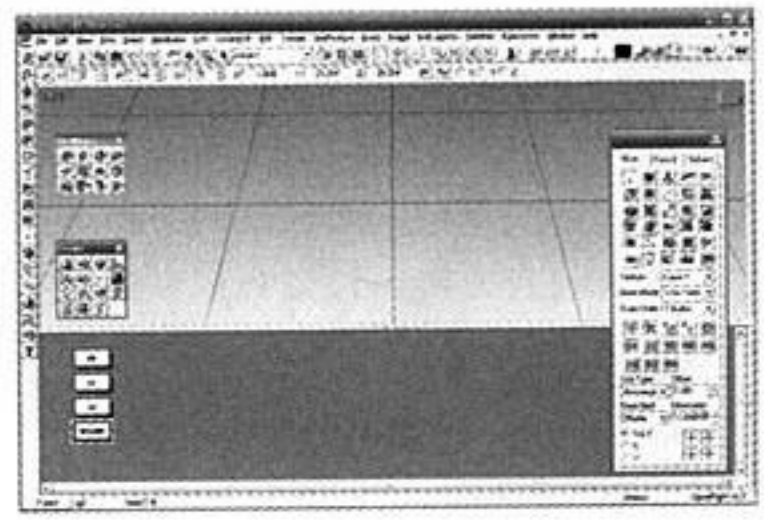


图 2 Creator 3.2 运行界面

打开 Creator,在默认状态下，创建一个外部引用节点，在外部引用对话框的“External File Name”编辑框中输入“tree01.flt”，单击“OK”按钮退出对话框。单击菜单“File”→“Save”，打开保存文件对话框，在文件名编辑框中输入“tree.flt”,单击“保存”按钮退出对话框，关闭 Creator。

3.3 编辑模型定义文件与交叉引用文件

可以用任意文本编辑器进行编辑，这里用的是 Windows 记事本，具体定义：

LAICAR.COM
shop35833438.taobao.com

编辑 vsgn_speedtree_defs.txt 文件内容如下：

```
MODEL tree01.speedtree {
    SOURCE_FILE    Beech_RT.spt
    INSTANCE_CULLING    false
    TECHNIQUE_DETAIL    true
    TECHNIQUE_SELF_SHADOW    true
    TECHNIQUE_DETAIL    true
    SCALE    1.0
}
```

编辑 vsgn_xref_map.txt 文件内容如下：

```
tree01.flt tree01.speedtree
```

这两个文件定义了外部引用文件和替换模型文件的对应关系。

如上所示，用模型 tree01.speedtree 代替模型 tree01.flt，而模型 tree01.speedtree 的源模型为文件 Beech_RT.spt，即最终 tree01.flt 被 Beech_RT.spt 代替。

3.4 编辑应用配置文件 ACF

应用配置文件（Application Configuration File，ACF）包含了 Vega Prime 应用在初始化和运行时所需的一切信息，采用流行的通用文件格式扩展标识语言（XML）格式存储，可以使用 Vega Prime 的可视化编辑器 LynX Prime 进行开发。这里使用的 LynX Prime 版本为 2.2，运行界面如图 3 所示。



图 3 LynX Prime 2.2 运行界面

ACF 文件具体定义步骤如下：

(1) 打开 LynX Prime，在默认状态下，创建一个 Object 实例，模型文件名选择前面用 Creator 创建的 tree.flt，模型位置设置为（0，60，55），其他参数默认。

(2) 创建一个 SpeedTreeControl 实例，参数设置默认。

(3) 编辑搜索路径 SearchPath，添加模型文件 tree.flt 和 Beech_RT.spt 所在的目录。

(4) 修改运动模式，将其设为 MotionGame。

(5) 修改观察者 myObserver 的位置（x，y，z）为（0，0，60），姿态（h，p，r）为（0，0，0）。

(6) 单击菜单“File”-“Save”，打开保存文件对话框，在文件名编辑框中输入“test.acf”，单击“保存”按钮退出对话框，关闭 LynX Prime。

3.5 程序运行效果

对上面建立的 ACF 文件，可以用 LynX Prime 打开，在 Active Preview 里进行预览，也可以用 VC 编程进行驱动。使用 VC 编程驱动时，可以动态改变环境的风速和风向，使 SpeedTree 模型做出随风摇摆的不同响应。在 Active Preview 里预览效果如图 4 所示，在 MFC 单文档框架下的运行效果如图 5 所示。



图 4 在 Active Preview 里预览的效果

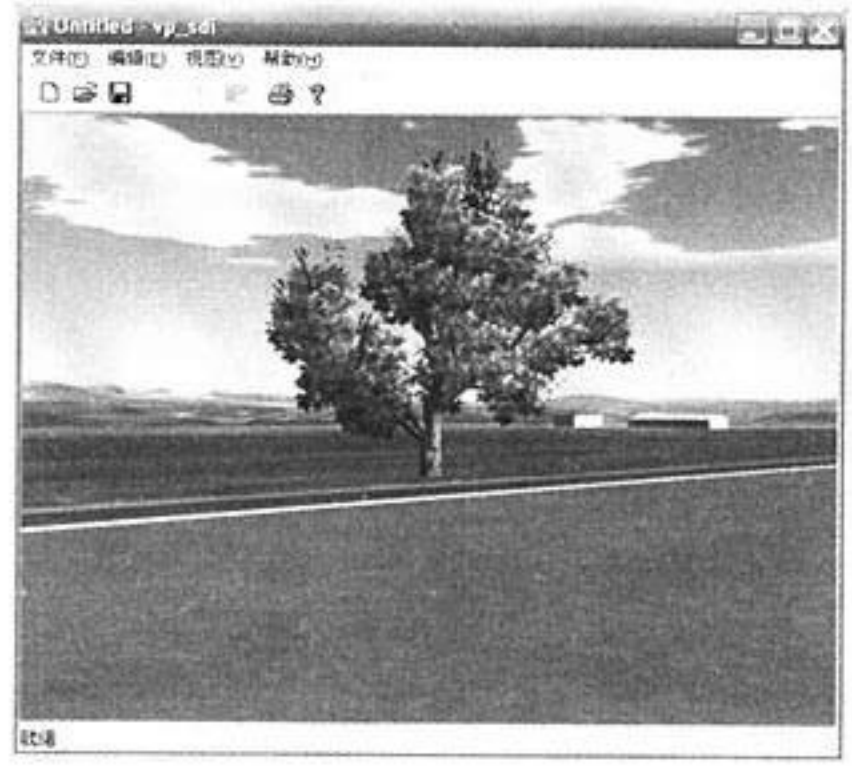


图 5 在 MFC 单文档框架下运行的效果

4 结语

SpeedTree 模块增强了 Vega Prime 中树木模型的真实性，调整 Vega Prime 里环境风的参数，SpeedTree 模型能够以摇摆的方式做出逼真的反应。结合实例介绍了 Vega Prime 中 SpeedTree 模块的基本程序设计方法，在此基础上可以开发 SpeedTree 模块的具体应用。程序的开发运行环境为：Windows XP SP3、VS2005、Vega Prime2.2、Creator3.2。

（收稿日期：2013-04-26）

来卡网出品
LAICAR.COM
shop35833438.taobao.com

基于 DI-Guy 的单兵游戏动作设计与实现

陶 阳

摘 要：介绍了 Vega 和 DI-GUY（人体运动仿真工具包）的基本知识，对单兵游戏动作进行了分类研究。设计并实现了几个简单的单兵游戏动作。单兵游戏动作逼真，动作切换自然，可以应用到单兵作战类游戏中。

关键词：Vega 工具；DI-Guy 软件；单兵作战；游戏开发

1 引言

单兵作战类游戏是广大游戏爱好者比较热衷的一类对抗型的游戏。游戏玩家在游戏中可以第一人称或第三人称的视角，控制游戏中的士兵对象，完成各种战术游戏动作，达到消灭敌人，保存自己的目的。游戏的对抗性较强，军事味儿比较重，这也是广大游戏爱好者喜欢此类游戏的原因之一。目前，单兵作战类游戏很多，比如大家最为熟悉的 CS 游戏、荣誉勋章、重返德军总部、抢滩登陆、战地系列游戏等。大部分游戏中，单兵模型都是 3D 的了，而且模型的精细度很高，视觉震撼较强，效果逼真，使得游戏玩家在玩游戏时有身临其境的感觉。对于普通的游戏开发者来说，最关心的就是游戏是如何设计的。作为单兵类游戏，单兵的游戏动作设计是比较关键的。

作为一款比较适用于军事视景仿真的软件开发工具，Vega 可以用于作战类游戏的开发。Vega 是一套完整地用于开发交互式、实时可视化仿真应用的软件平台，其最基本的功能是驱动、控制、管理虚拟场景并支持快速复杂的视觉仿真程序，快速创建各种实时交互的三维环境，快速建立大型沉浸式或非沉浸式的虚拟现实系统。Vega 中进行人物仿真基本上可以分为 3 种方式：借助 Multigen Creator 中所附带的动画序列（Animation Sequence）技术产生动画效果；借助多种建模工具，生成基于骨架、蒙皮的人物模型，然后通过软件之间的接口将人物模型转化为 Vega 能够识别的模型结构，然后进行仿真；通过专用软件进行仿真。由于第一种方法所采用的是模拟传统二维动画创作的方法，通过连续播放一系列动画画面来产生人物连续运动的效果，在 Creator 中就是通过按照指定的顺序依次显示组节点下的单个子节点来实现这个过程。但在复杂的仿真过程中，当对动作的频率、样式等要求较高时，此种人物仿真就显得过于单调，失去了真实性。MultiGen Creator 是对大地形场景建模最好的工具，这同时也决定了它对于人物建模这样细节要求较高的建模方式并不擅长，因此需要其他软件的辅助，才能够很好地进行人物建模。3D Max 虽然对细节建模较为擅长，但由于人体的体构造的复杂性，对于非艺术建模出身的人来说建立精确而美观的人体模型仍是一件很困难的事情，因此也不适合快速进行人体建模。使用 DOF 节点进行人物仿真，需要对人物模型添加并设置多个 DOF 节点，且人物的仿真程度越高、效果越好所需要的 DOF 节点数就越多，在 Vega 中进行 DOF 节点控制时，所需要编写多 DOF 节点之间的协调控制算法就越为复杂，而且需要大量的测试才能达到用户所需效果，既费时又费力。对一般游戏开发者来说，难度大，学习起来较为困难。

所以，这里基于 Vega 软件开发环境，通过专用软件（Boston Dynamics 公司的 DI-Guy 软件）进行仿真，完成单兵作战类游戏中的单兵游戏动作的设计与实现。

2 DI-Guy 概述

波士顿动力（Boston Dynamics）公司的 DI-Guy 软件，是以 OpenFlight 格式为基础的三维人物仿真软件，可以完全集成到 Vega 工具包中。每个“人物”可以根据简单的指令在环境中漫游。DI-Guy 自动地给人物提供相应的特征，而不需要动画师的处理。甚至在人物间的转换中，也可以提供自然的运动和无缝的转化。用户还可以通过 Vega 中的 LynX 控制面板和 DI-Guy API 定义特别的行为。例如：人物的衣服和装备、被赋予的行为等。

DI-Guy 是全程交互的软件，可以响应所有实时的指令。软件通过运动缓存优化、可变动作差补、高水平细节的转化、运动细节的转化、任务水平的控制等来达到逼真的实时效果。这些技术使同时显示几十种连接的 DI-Guy 特征成为可能。DI-Guy 最初是为美国军方进行兵力仿真而进行设计的，其中包括一套描绘步兵特征的选项。它利用了真实士兵训练的特征，具有达 7 级细节水平的全纹理模型、多种制服（战斗服、沙漠伪装、陆战服、伊拉克绿等）、武器（M16、AK47、M203）和多种附属装备（背包、水壶、刺刀等）。它可定义的行为包括：站、跪、匍匐前进、走、跳、潜行、使用武器及其他活

LAICAR.COM
shop35833438.taobao.com

动。除此之外，DI-Guy 也包括一些其他种类人物及其相关动作，这使得使用变得非常简单。

DI-Guy 软件动作类型及人物的模型相当丰富。基于 DI-Guy 的虚拟人运动控制主要是利用 DI-Guy 强大的 API 函数，其虚拟人的运动动作非常逼真，能够产生较好的沉浸感。DI-Guy 很好地将基于 OpenFlight 格式的人物及其动作进行了封装，以面向对象的方式提供给了用户 API 函数库，使得用户不需要进行大量的 DOF 节点的运动协调操作就可以在 Vega 中直接调用其 API 函数进行人物动作的控制与仿真。但 DI-Guy 不是作为 MultiGen Vega 的一部分，需要单独购买安装，才能与 Vega 结合。将 DI-Guy 安装后并添加相应的许可证文件后，将在 Vega 的 LynX 控制面板的左侧选项卡中出现 DI-Guy 的模块选项，如图 1 所示。



图 1 Vega 的 LynX 控制面板的 DI-Guy 模块选项

这样，在 LynX 中添加 DI-Guy 的人物，在仿真过程中采用相应的 API 函数就可实现在 Vega 中对 DI-Guy 人物进行控制了。除了在 LynX 中添加 DI-Guy 功能之外，大多数游戏开发者更喜欢在程序代码中直接使用 API 函数来实现对人物对象的控制。

3 单兵游戏动作的设计与实现

单兵的几个常用的游戏动作分类如图 2 所示。



图 2 几个常用的单兵游戏动作

常用的单兵的游戏动作主要包括持枪、卧倒、屈身行进、匍匐前进、跃进等 5 类。持枪是单兵在战斗中为了便于行动、便于射击携带武器的方法。在不同的地形和距离条件下，根据敌情和任务特点，采用不同的持枪动作。卧倒包括起立动作，是单兵遭敌突然袭击时或袭击后采取的动作，以躲避敌人的火力打击。屈身行进包括直身行进，在单兵通常运动的姿势，可根据距离敌人的远近以及地形条件，灵活地直身前进或屈身前进。屈身前进通常是在遮蔽物略低于人体时采用。其要领是：右手持枪，上体前倾，两腿弯曲，屈身程度视遮蔽物高低而定，目视前方，以大步或快步前进。匍匐前进是通过敌火力封锁下的较短地段或利用较低遮蔽物时采用。根据遮蔽物的高低分为低姿、高姿、侧身和高姿侧身匍匐 4 种。跃进是在敌火下迅速通过开阔地时经常采用的运动方法。其要领是：跃进前，可左、右移动（滚动）以迷惑敌人，迅速收枪，屈左脚于右腿下，右手提枪，以左手、左膝、左脚的力量将身体撑起，迈出右脚，突然跃起前进。也可在收枪的同时，屈左腿于腹下，以左手、右膝和左小腿的外侧支撑身体，迈出右脚，突然跃起前进。跃进时，右手持枪，目视敌方，屈身快跑。跃进的距离、速度应根据敌火力强弱和地形情况而定。地形越开阔，敌火力越猛烈，跃进的距离应越短，速度应越快。

根据以上分析，给出实现单兵游戏动作的系统的功能设计图，如图 3 所示。系统初始化是单兵游戏动作模拟系统的环境的初始化，是对 Vega 和 DI-Guy 模块的初始化，是对 DI-Guy 中的士兵模型对象的引入并使其就位的一个过程。初始化之后的士兵，在仿真环境中就可以用来进行各种单兵游戏动作的测试了。持枪瞄准模块，就是士兵进行持枪动作的一个过程演示，士兵经过携枪行进、举枪瞄准、收抢收步等几个环节。屈身行进是士兵对象直身行进，然后屈身行进，再恢复直身行进并返回的一个过程。匍匐前进与跃进动作，同屈身行进类似，作为单兵游戏动作模拟系统来说，这些仅仅是对单兵游戏动作的过程演示，以观察其动作衔接是否自然，模拟过程是否逼真。系统退出即是退出游戏动作模拟系统。

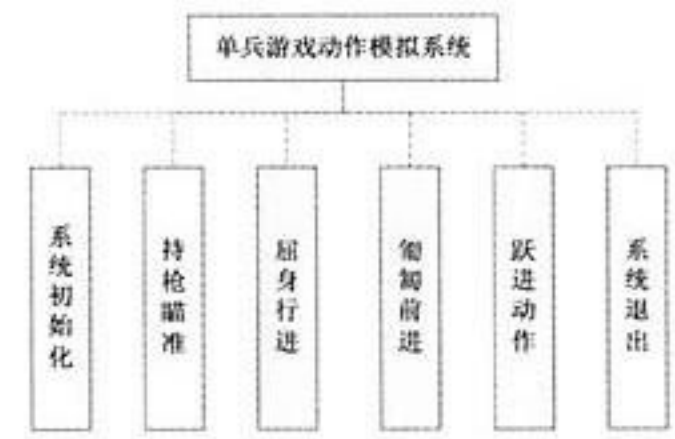


图 3 单兵游戏动作模拟系统功能结构图

在 VC++60 开发环境中，首先设置好 Vega 和 DI-Guy 的包含路径，然后建立工程 SoldierTactics。当然，在此之前，为了加快开发应用程序的速度，首先在 Vega 提供的 LynX 图形界面下，配置好一个 ADF 文件 vegawin.adf，包括地形数据、纹理数据、天空环境等。配置好的 vegawin.adf 预览的效果如图 4 所示。



图 4 配置好的自然环境仿真效果图

来卡网出品
LAICAR.COM
shop35833438.taobao.com

也可以在 LynX 环境下借助 DI-Guy 模块加入士兵对象。但考虑到使用程序代码，通过控制 DI-Guy 的 API 来控制士兵对象会更加灵活，这里仅给出通过 LynX 加入人物对象的一般方法步骤，如下：

(1) 在 Vega 中建立一条路径，设置为 path1，文件名 1.pth。

(2) 在 LynX 面板中创建一个新的导航器，命名为 Nav1，其文件名 1.nav，使之与 path1 建立关联。

(3) 打开 Vega 的 Path 工具，选中 Nav1，设定其类型为 BDI Character，然后选择 DI-Guy 的士兵，并添加装备。在视图区鼠标点击关键点，生成一条路径，再选择人物动作。动作可分为两种，一种是在关键点停留时的动作，如原地瞄准等；另一种是在关键点间移动的动作，如走、跑等。

(4) 将路径和导航器进行存储，在 Isector 面板中生成一个新的 Isector，类型为 LOS，目标为场景，Isector 的类别定义为 Terrain 型。

(5) 在 Vega DI-Guy 面板生成一个 DI-Guy 人物，命名为 Soldier1，使其关联到 Nav1。

(6) 在场景中加入 Soldier1，然后在应用程序的代码中包含头文件 vgdiguy.h 及 DI-Guy 角色头文件 atlas.h。

接下来的应用程序实现就跟 API 调用的方法一样了。下面给出具体的主要功能模块的实现算法。系统初始化部分实现的主要代码如下：

```
void CVegaOptions::OnForward()
{
 //初始化 DI-guy,引入我们要控制的士兵
   vgWindow*  win=vgGetWin(0);
 vgObserver  *mainObs = vgGetObserv( 0 );
 {
      vgChannel * chan = vgGetChan( 0 ); /* 获得第一个通道 */
                vgAddFunc  (chan, VGCHAN_PREDRAW, update_diguy_characters, NULL);
 }
 SetTimer(818, 3000, NULL);
}
//DI-Guy 对象的更新
void update_diguy_characters(void *handle, void *tmp)
{
  static int n = 0 ;
  static float orientation = -90.0f;
  float dt = (float) vgGetDeltaFrameTime();
  if (0 == n)
  {
       /* 初始化 */
       bdi_log_stdout_enable(BDI_LOG_WARN);
       bdi_log_file_enable   (BDI_LOG_DEBUG,   "simple.log", 1);
       if (diguy_initialize_pf(NULL)) {
            bdi_log_printf        (BDI_LOG_WARN,        "diguy_initialize_pf failed! \n");
            exit ( EXIT_FAILURE ) ;
       }
       /* 创建人物对象 */
       {
       static char* example_custom_equipment_list[] = {
            "soldier_black:torso", /* replace bdu torso with black */
            "lw_equip:canteen2", /* add a canteen */
            "lw_equip:mic", /* add a mouthpiece*/
            "lw_m4c", /* replace m16 with m4c */
            "lw_m4c_flash", /* add a muzzle flash */
       "lw_m4c_attachments:bay", /* add a bayonet */
            "bdu",
            NULL
       };
            /创建士兵对象
              soldier = create_diguy_character("soldier2", example_custom_equipment_list,
            2615.0f, 2444.0f, 0.0f, 180.0f+orientation, 1.0f, "stand_ready");/
       }
       /* 将士兵对象加入到场景 */
       {
            vgScene * scene_vg = vgGetScene(0);
            pfScene  *  scene_pf  =  vgGetScenePfScene (scene_vg);
            diguy_set_graphics_attach_ptr (DIGUY_ALL_CHARACTERS_HANDLE,
                 (void *) scene_pf);
       }

  }
    //DI-Guy 更新
  diguy_update (DIGUY_ALL_CHARACTERS_HANDLE, dt) ;
   return ;
  }
```

持枪瞄准的功能实现代码如下：

```
void CVegaOptions::OnHanggun()
{
  //首先让士兵行走
   diguy_set_action(soldier,diguy_query_action_token_by_name   (diguy_query_character_type_by_name   ("soldier2"),"walk"));
  //设置定时器,控制士兵瞄准的动作
  SetTimer(2, 5000, NULL);
}
```

另外 3 个功能模块的实现代码同持枪瞄准，只是它们设置的定时器不同，因为，在设计单兵动作的时候，考虑到动作之

LAICAR.COM shop35833438.taobao.com

间切换的自然性，都是从行进（walk）状态开始的，具体的DI-Guy的控制都在定时器函数里面实现了，通过定时功能，实现士兵具体动作的连续。所以，这里的定时器的实现很重要，具体的代码如下：

```
void CVegaOptions::OnTimer(UINT nIDEvent)
{
float x, y, z, h, p, r,ang;
if (nIDEvent == 818)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{
x=2615.0f;y=2444.0f;z=0.0f;
diguy_set_desired_position(soldier,&x, &y, &z);//设置目的地
}
//取消定时器
KillTimer(818);
SetTimer(1, 2000, NULL);
UpdateData(FALSE);
}
if (nIDEvent == 886)
{
//取消定时器
KillTimer(1);
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{
//设置士兵状态站立准备
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"stand_
ready"));
}
//取消定时器
KillTimer(886);
UpdateData(FALSE);
}
if (nIDEvent == 888)
{
//取消定时器
KillTimer(1);
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{
//设置士兵状态站立准备
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"stand_ready"));
diguy_get_position(soldier,&x, &y, &z, &h);
//设置士兵状态行进
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"walk"));
diguy_set_orientation(soldier, h+180.0f);
}
SetTimer(889, 7000, NULL);
//取消定时器
KillTimer(888);
UpdateData(FALSE);
}
if (nIDEvent == 889)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{
diguy_get_position(soldier,&x, &y, &z, &h);diguy_set_
action(soldier,diguy_query_action_token_by_name(diguy_qu
ery_character_type_by_name("soldier2"),"stand_ready"));
diguy_set_orientation(soldier, h+180);
}
//取消定时器
KillTimer(889);
UpdateData(FALSE);
}
if (nIDEvent == 1)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"walk"));
//设置定时器
SetTimer(886, 8000, NULL);
}
//取消定时器
KillTimer(1);
UpdateData(FALSE);
}
if (nIDEvent == 2)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{//设置士兵站立瞄准
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"stand_aim"));
//设置定时器
SetTimer(888, 3000, NULL);
}
//取消定时器
KillTimer(2);
UpdateData(FALSE);
}
if (nIDEvent == 3)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
{//设置士兵低姿行进
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"walk_lo"));
//设置定时器
SetTimer(33, 5000, NULL);
```


来卡网出品
LAICAR.COM
shop35833438.taobao.com

```
                    }
                //取消定时器
                KillTimer(3);
            UpdateData(FALSE);
    }
    if (nIDEvent == 33)
    {
    if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
                {
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"walk"));
                        //设置定时器
                SetTimer(333, 2000, NULL);
                  }
                  //取消定时器
                  KillTimer(33);
                UpdateData(FALSE);
    }
    if (nIDEvent == 333)
    {
    if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
                {
          diguy_get_position(soldier,&x, &y, &z, &h);
                    diguy_set_orientation(soldier, h+180);
                    //设置定时器
                SetTimer(3333, 8000, NULL);
                  }
                  //取消定时器
                  KillTimer(333);
                UpdateData(FALSE);
    }
    if (nIDEvent == 3333)
    {
          if (soldier != DIGUY_BAD_CHARACTER_HANDLE)
                {
                diguy_get_position(soldier,&x, &y, &z, &h);
                    diguy_set_orientation(soldier, h+180);
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"stand_re
ady"));
                    }
                //取消定时器
                  KillTimer(3333);
                UpdateData(FALSE);
    }
    if (nIDEvent == 4)
    {
  if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
                {//设置士兵爬行
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"crawl"));
                    //设置定时器
                SetTimer(44, 5000, NULL);
                  }
                  //取消定时器
                  KillTimer(4);
                UpdateData(FALSE);
    }
    if (nIDEvent == 5)
    {
    if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
                {//设置士兵跳步
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"jog"));
                    //设置定时器
                SetTimer(55, 5000, NULL);
                  }
                  //取消定时器
                  KillTimer(5);
                UpdateData(FALSE);
    }
    if (nIDEvent == 55)
    {
    if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
                {
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"crawl"));
                    //设置定时器
                SetTimer(5551, 5000, NULL);
                  }
                  //取消定时器
                  KillTimer(55);
                UpdateData(FALSE);
    }
    if (nIDEvent == 5551)
    {
          if (soldier != DIGUY_BAD_CHARACTER_HANDLE)
                {
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"walk"));
                    //设置定时器
                SetTimer(555, 5000, NULL);
                  }
                  //取消定时器
                  KillTimer(5551);
                UpdateData(FALSE);
}
if (nIDEvent == 555)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
                {
                diguy_get_position(soldier,&x, &y, &z, &h);
                    diguy_set_orientation(soldier, h+180);
                    //设置定时器
```


LAICAR.COM
shop35833438.taobao.com

```
            SetTimer(5555, 13000, NULL);
              }
              //取消定时器
              KillTimer(5555);
            UpdateData(FALSE);
}
if (nIDEvent == 5555)
{
if (soldier ! = DIGUY_BAD_CHARACTER_HANDLE)
              {
          diguy_get_position(soldier,&x, &y, &z, &h);
                    diguy_set_orientation(soldier, h+180);
diguy_set_action (soldier,diguy_query_action_token_by_name
(diguy_query_character_type_by_name("soldier2"),"stand_re
ady"));
              }
              //取消定时器
              KillTimer(5555);
            UpdateData(FALSE);
    }
   }
```

系统编译运行后的效果如图 5 所示。左边是单兵游戏动作控制按钮，右边是一个虚拟的场景，士兵的动作就是在这个场景中进行演示的。



图 5　单兵游戏动作模拟系统运行效果图

点击“初始化”按钮后，一个用于测试动作的士兵对象就会慢步走入到场景中，如图 6 所示。



图 6　系统初始化效果图

点击“持枪瞄准”、“屈身行进”、“匍匐前进”、“跃进动作”按钮之后的效果分别如图 7、图 8、图 9、图 10 所示。从结果可以看出，这几个主要的单兵游戏动作连贯性很好，动作之间衔接比较自然，与场景一起，模拟效果逼真。



图 7　持枪瞄准效果图



图 8　屈身行进效果图



图 9　匍匐前进效果图



图 10　跃进动作效果图

4　结语

基于 DI-Guy 的三维人物模型的运动及显示开发包可以广泛应用于仿真等领域，它具有功能强大、使用方便、可扩展性

来卡网出品 LAICAR.COM shop35833438.taobao.com

强等优点。在设计并实现的单兵游戏动作展示系统中，其人物模型还比较精细，单兵的游戏动作关联衔接自然，逼真度很高，可以广泛应用于需要人物交互功能的仿真领域（如作战类游戏开发中）。

随着 DI-Guy 工具包的完善以及应用力度的加大，需要进一步解决的问题还有很多，例如新增士兵模型到 DI-Guy 模型库、在程序中实现对人物表情、小动作等灵活控制等，这也是下一步展开深入研究的工作。

参考文献

[1] 李石磊，梁加红，柏友良，陈凌. DI-Guy 中人体动作生成方法研究. 计算机仿真，2008，25（10）.

[2] 周俊峰，祝奎臣. 船舶机舱虚拟环境中的虚拟人构建. 上海海事大学学报，2007，28（4）.

[3] 谢建华，马立元，张睿，侯雷. 基于 DI-Guy 的某型导弹虚拟操作训练环境设计. 计算机仿真，2005，22（3）.

[4] 方琦峰。康凤举，王宝龙，王娇艳. 基于 DI-Guy 的人体视景仿真研究与应用. 计算机工程与科学，2008，30（1）.

[5] 刘良，关正西. 基于虚拟人的模拟操作训练研究. 计算机工程与设计，2005，26（10）.

[6] 陈喜春. 基于 DI-Guy 和 VR-Link 的分布交互仿真的实现. 计算机应用与软件，2011，5.

[7] 张伟. 矿井虚拟避灾系统的研究与实现 [D]. 山东科技大学，2008.

（收稿日期：2013-03-09）

（上接第 68 页）

件完全一样，则表示程序正确，去除运行答案的每行末尾的空格，回车，还发现其他字符的不匹配，立即中止该程序，修改数据库记录为 Wrong Answer，用户可以看到自己的程序的正确性。

4.3.3 评判结果准确性

为了防止学生针对答案造程序的可能性，本设计通过程序产生多组随即输入数据 (*.in) 输出数据 (*.out)，对提交程序进行多次验证。

4.4 管理模块

管理员可以向题库中添加各种类型且符合要求的试题,也可以对它们进行修改和删除。同时，管理员也能对用户、编程竞赛和 BBS 等数据进行管理。

（1）题目管理

①题目录入，管理员可以手工输入也可以从文件导入。②试题修改，管理员还可以对试题进行修改。③试题删除，管理员可以删除不再需要的试题。

（2）竞赛管理：管理员可以随时抽取题目规定时间安排编程竞赛。

4.5 BBS 模块

程序设计方法的提高需要大家探讨，这就是这个论坛建设的必要性。BBS 设计采用树形结构，可以通过题号进行索引浏览，使问答层次更分明。

4.6 数据库

根据功能需求建立 judgeonline 数据库，主要包括 user 用户、problem 题目、solution 提交答案、message 发布信息、contest 竞赛、Cont-est_problem、news 等数据库表。

User 表字段主要有:user_id、submit、language、password、nick、school 等。

problem 表字段主要有:problem_id、title、description、input、output、hint 、source、date、time_limited、acceptec 等。

solution 主要字段有:solution_id、problem_id、user_id、time、date、result、language、contest_id、code_length 等。

message 表用与 BBS 讨论模块,主要字段有：message_id、problem_id、parent_id、depth,user_id、title、content、date 等。

5 结语

系统通过 PHP，MySQL，Apache 组合实现了用户注册、更新、提交、查看状态和 BBS 讨论等功能，对 C，C++程序进行了成功编译运行。在线评测系统充分利用网络的快捷、方便，为提高同学编程能力提供了理想的平台，评测系统将会得到广泛应用。

参考文献

[1] 张恩民，编著. PHP 开发实战权威指南 . 清华大学出版社，2012.

[2] POJ. 北京大学在线评测系统. http:acm.pku.edu.cn/JudgeOnline/.

[3] [美] George Reese，等著，林琪，等译. MySQL 权威指南. 中国电力出版社，2003.

（收稿日期：2013-01-31）

LAICAR.COM
shop35833438.taobao.com

C# 语言实现从 Word 文档中提取文本

吴春龙

摘　要：从 Word 文档二进制文件数据中，过滤掉 Word 文档中的各种格式控制等数据，提取 Word 文档中用户需要的文本。这是一款非常实用的从 Word 文档中提取文本工具软件。

关键词：C# 语言；Word 文档；提取文本；工具软件

1　引言

前段时间，进行数据恢复软件开发，需要将保存在磁盘上的二进制文件数据，按照不同的文件格式，进行相应的文档（数据）恢复。数据恢复时，如果数据区的数据连续并且没有破坏，则恢复文件比较容易；如果数据区的数据损坏，恢复文件就十分困难。Word 文档由系统产生的标准格式控制数据和用户数据组成，不同版本的 Word 文档，例如 Word97、Word2003、Word2007、Word2010 等，系统产生的标准格式控制数据和编码方式都不同。该工程实例采用作者在进行数据恢复时开发的 ReadWordTextClass.dll 动态链接库，进行软件设计，实现了从 Word97、Word2003 文档中提取文本。为了检验该工具软件的实际效果，与国外专业的从 Word 文档中提取文本的工具软件作了测试比较，测试结果：从 Word 文档中提取文本工具软件性能优于专业软件。

2　ReadWordTextClass.dll 动态链接库函数

（1）获得 Word97 文档第一段文本：

```
public bool GetWord97FirstString(long StartS,//文件开始指针
                                 long FsLength, //文件长度
                          BinaryReader r,//用特定的编码将基元数据类型读
//作二进制值
                                 int NoNum, //段标号
                                 out long NewS,// 获得文件指针
                           out string StrUnicode//获得文本
                                 )
```

（2）获得 Word2003 文档第一段文本：

```
public bool GetWord2003FirstString(long StartS, //文件开始指针
                                   long FsLength, //文件长度
                            BinaryReader r, //用特定的编码将基元数据类型读
//作二进制值
                                   out bool ZsetS, // Word 文档标志
                                   int NoNum, //段标号
                                   out long NewS, //获得文件指针
                                   out string StrUnicode//获得文本
                                   )
```

（3）获得 Word 文档文本：

```
public bool GetWordString(long StartS, //文件开始指针
                          long FsLength, //文件长度
                          int SetNum,//设置连续获得有效文本数
                           BinaryReader r, //用特定的编码将基元
//数据类型读作二进制值
                          bool ZsetS, // Word 文档标志
                       int StartNum,// 设置文件指针
                          int NoNum, //段标号
                          out long NewS, //获得文件指针
                          out string StrUnicode//获得文本
                         )
```

（4）连续获得 Word 文档文本：

```
public void GetWordSecondlyString16(long StartS, //文件开始
//指针
                                    long FsLength, //文件长度
                                BinaryReader r, //用特定的编码将基元
//数据类型读作二进制值
                                int SetNum, //设置连续获得有效文本数
                                   bool ZsetS, // Word 文档标志
                                      int NoNum, //段标号
                                out string StrUnicode//获得文本
                                      )
```

3　实现代码（软件开发平台 C#2008）

（1）新建一个 Windows 窗体应用程序工程。在 Form1 窗体上分别放入控件 progressBar1、richTextBox1、menuStrip1、openFileDialog1、timer1、saveFileDialog1，在工程目录下拷入文件 ReadWordTextClass.dll。

（2）添加引用。在解决方案资源管理器中添加引用 ReadWordTextClass.dll。

（3）文件菜单。选中 menuStrip1，设置文本为“文件”，在“文件”下分别创建子菜单“打开 *.doc 文件”、“保存为 rtf 文件格式”、“保存为 txt 文件格式”。

（4）在 Form1.CS 文件中添加引用：

来卡网出品 LAICAR.COM shop35833438.taobao.com

```
using System.Threading;//add
using System.IO;
using ReadWordTextClass;
```

(5) 在 Form1 中添加下列语句和函数：

```
//调用 C# 创建的动态链接库
Class1 myClass1 = new Class1();
// 定义委托
private delegate void Mydelegate(string str);
private delegate void MydelegateClear();
 private delegate void MydelegateInt (long i, long fsLength);
//进度条参数设置
int progressBar1Value = 0;
int progressBar1Maximum = 100;
//richTextBox1 线程清空函数
private void richTextBox1ShowClear()
{
    if (this.InvokeRequired)
    {
                MydelegateClear mydelegate = new MydelegateClear(richTextBox1ShowClear);
        Invoke(mydelegate);
    }
    else
    {
        //在窗体上显示文件路径和文件名
        this.Text =" 从损坏的 Word 文件中提取文本 "+ openFileDialog1.FileName;
        //清空
        richTextBox1.Text = "";
    }
}
//richTextBox1 线程显示字符串函数
private void richTextBox1ShowStr(string str)
{
    if (this.InvokeRequired)
    {
            Mydelegate mydelegate = new Mydelegate (richTextBox1ShowStr);
        Invoke(mydelegate, str);
    }
    else
    {
        richTextBox1.Text += str;
        richTextBox1.Text = richTextBox1.Text.Trim();
    }
    return;
}
//控制进度条和 richTextBox1 函数
private void ShowInt(long i, long fsLength)
{
    if (this.InvokeRequired)
    {
MydelegateInt mydelegateInt = new MydelegateInt(ShowInt);
        Invoke(mydelegateInt, i, fsLength);
    }
    else
    {
        //转换结束关闭进度条,开启 richTextBox1 显示
        if (i >= fsLength - 1)
        {
            progressBar1.Visible = false;
            richTextBox1.Visible = true; ;
        }
    }
}
//定时器事件代码
private void timer1_Tick(object sender, EventArgs e)
{
    if (progressBar1Value < progressBar1Maximum)
    {
        //进度
        progressBar1.Minimum = 0;
 progressBar1.Maximum = progressBar1Maximum;
        progressBar1.Value = progressBar1Value;
        progressBar1Value++;
    }
    else
    {
        progressBar1Value = 0;
    }
}
//线程函数
 void ThreadProcWordAll()
{
        FileStream fs;
        BinaryReader r;
        long FsLength;
        try
        {
            fs = new FileStream (openFileDialog1.FileName, FileMode.OpenOrCreate);
            FsLength = fs.Length;//文件长度
            r = new BinaryReader(fs);
        }
        catch (Exception e)
        {
            //关闭计时器
            timer1.Stop();
MessageBox.Show("文件已经在别处打开,请关闭该文件！");
            return;
        }
        //清空 richTextBox1
        richTextBox1ShowClear();
```


shop35833438.taobao.com

```
        int NoNum = 1;//标号
        int SetNum = 0x10;//设置连续获得有效字符的个数
        long NewS = 0;
        long StartS = 0;//作为读取文件指针
        int setS = 0;//设置指针
        bool ZsetS = false;
        string StrUnicode = "";
        if (myClass1.GetWord97FirstString(StartS, FsLength,
r, NoNum, out NewS, out StrUnicode))
        {
            NoNum++;
        }
        if (StrUnicode.Length == 0x4e)
        {
            //字符转 ASCII
             byte [] bytesASCII = Encoding.ASCII.GetBytes
(StrUnicode.Substring(0x4e-1,1));
            if (bytesASCII[0] == 0xd)
            {
                richTextBox1ShowStr(StrUnicode);
                goto endRUN;
            }
        }
        richTextBox1ShowStr(StrUnicode);
        StrUnicode = "";
        StartS = NewS;
          if    (myClass1.GetWord2003FirstString   (StartS,
FsLength, r, out ZsetS, NoNum, out NewS, out StrUnicode))
        {
            NoNum++;
        }
        else
        {
            if (ZsetS)
            {
                richTextBox1ShowStr(StrUnicode);
                goto endRUN;
            }
        }
        StartS = NewS;
        richTextBox1ShowStr(StrUnicode);
        if (ZsetS&&(StrUnicode.Length < 128))
        {
            goto endRUN;
        }
        StrUnicode = "";
        setS = 0x11fa;
        if (ZsetS && (StartS < setS))
        {
            NewS = 0;
              myClass1.GetWordString   (StartS,   FsLength,
SetNum, r, ZsetS, setS, NoNum, out NewS, out StrUnicode);
            if (StrUnicode ! = "")
            {
                NoNum++;
            }
            StartS = NewS;
        }
        if (ZsetS && (NoNum == 3))
        {
            richTextBox1ShowStr(StrUnicode);
        }
        else
        {
            StrUnicode = "";
            setS = 0x13fa;
            if (ZsetS && (StartS < setS))
            {
                NewS = 0;
                 myClass1.GetWordString  (StartS,  FsLength,
SetNum, r, ZsetS, setS, NoNum, out NewS, out StrUnicode);
                if (StrUnicode ! = "")
                {
                    NoNum++;
                }
                StartS = NewS;
                if (ZsetS && (NoNum == 3))
                {
                    richTextBox1ShowStr(StrUnicode);
                }
                else
                {
                    StrUnicode = "";
                     myClass1.GetWordSecondlyString16  (StartS,
FsLength, r, SetNum, ZsetS, NoNum, out StrUnicode);
                    richTextBox1ShowStr(StrUnicode);
                }
            }
        endRUN: ;
            //关闭计时器
            timer1.Stop();
            //关闭文件
            r.Close();
            fs.Close();
            //关闭进度条 显示文本
            ShowInt(FsLength, FsLength);
            return;
        }
```

（6）打开 *.doc 文件菜单事件代码：

```
        private void toolStripMenuItem11_Click (object sender,
EventArgs e)
        {
            openFileDialog1.CheckFileExists = true;
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

```
        openFileDialog1.Multiselect = false;
        openFileDialog1.Filter = "文件类型(*.doc)|*.doc|所有
文件(*.*)|*.*";
        openFileDialog1.Title = "请选择一个格式文件";
        openFileDialog1.InitialDirectory = "C:";
        openFileDialog1.ShowReadOnly = true;
        openFileDialog1.ReadOnlyChecked = true;
        openFileDialog1.ShowHelp = true;
    if ((openFileDialog1.ShowDialog() == DialogResult.OK)
            & (openFileDialog1.FileName.Length > 0))
        {
    if (this.openFileDialog1.ReadOnlyChecked == true)
            {
                this.richTextBox1.ReadOnly = true;
            }
            else
            {
                this.richTextBox1.ReadOnly = false;
            }
            //创建线程
            Thread thread = new Thread (new ThreadStart
(ThreadProcWordAll));
            //设置线程为后台
            thread.IsBackground = true;
            //打开进度条
            progressBar1.Visible = true;
            //关闭 richTextBox1
            richTextBox1.Visible = false;
            //置进度条初值
            progressBar1.Value = 0;
            //启动线程
            thread.Start();
            //启动计时器
            timer1.Start();
        }
    }
```

(7) 保存为 rtf 文件格式菜单事件代码：

```
    private void 保存为 rft 文件 ToolStripMenuItem_Click
(object sender, EventArgs e)
    {
    try
    {
        saveFileDialog1.AddExtension = true;
        saveFileDialog1.DefaultExt = "rtf";
        saveFileDialog1.CheckPathExists = true;
        saveFileDialog1.Filter = "格式文件(*.RTF)|*.rtf";
        saveFileDialog1.OverwritePrompt = true;
if ((saveFileDialog1.ShowDialog() == DialogResult.OK)
            & (saveFileDialog1.FileName.Length > 0))
        {
            //将 richTextBox1 中的字符写入文件
 richTextBox1.SaveFile(saveFileDialog1.FileName);
        }
    }
    catch
    {
        MessageBox.Show("保存文件失败！");
    }
    return;
}
```

(8) 保存为 txt 文件格式菜单事件代码：

```
                                    private void 保存为 txt 文件格式
ToolStripMenuItem_Click(object sender, EventArgs e)
    {
        try
        {
            saveFileDialog1.AddExtension = true;
            saveFileDialog1.DefaultExt = "TXT";
            saveFileDialog1.CheckPathExists = true;
            saveFileDialog1.Filter = "格式文件(*.TXT)|*.txt";
            saveFileDialog1.OverwritePrompt = true;
    if ((saveFileDialog1.ShowDialog() == DialogResult.OK)
                & (saveFileDialog1.FileName.Length > 0))
            {
                //将 richTextBox1 中的字符写入文件
                    richTextBox1.SaveFile    (saveFileDialog1.
FileName, RichTextBoxStreamType.PlainText);
            }
        }
        catch
        {
            MessageBox.Show("保存文件失败！");
        }
        return;
    }
```

4 以 Word2003 文档为例测试

(1) Word2003 文档正常情况测试（信息化战争.doc）

图 1 是该工具软件测试 Word2003 文档正常时能够提取 Word 文本。



图 1 该工具软件测试 Word2003 文档正常时能够提取 Word 文本

（下转第 93 页）

LAICAR.COM
shop35833438.taobao.com

XPE 嵌入式设备的远程在线调试

仲光亮

摘 要：XPE 架构是研发嵌入设备时可选择的重要平台架构，可以使用微软的多数应用程序和开发环境进行研发，研发人员入手快。针对此描述了实现远程的在线调试，以期更加高效地进行研发活动，调试解决嵌入式设备的问题。

关键词：XPE 架构；远程调试；Visual Studio 平台

1 XPE 嵌入式开发环境概述

XPE 是微软嵌入式产品线中的重要成员，目前命名为 Windows Embedded Standard，其本质是一个高度可定制的组件化 XP Professional 操作系统。为适应机顶盒、ATM、工控设备等嵌入式产品的定制化要求，将操作系统拆成多达上万个组件，不同的应用可选择定制，以得到更简洁健壮的嵌入式系统。其能运行在任何 X86 架构的处理器上，为台式机开发的调用 WIN32API 的应用程序也可运行在 XPE 环境平台上（所需组件已添加到 XPE 环境中），若.Net 框架组件已加入 XPE，XPE 也可运行.Net 程序。XPE 可利用熟悉的集成开发工具如 Visual Studio 等来高效研发强大的嵌入式应用程序。

Visual Studio 是一套完整的应用程序开发集成环境，其中包括与应用程序相关的各种分析与建模工具。XPE 开发环境包含目标设计器、组件设计器、组件数据库管理器等工具。组件设计器能够选择组件并添加到运行时配置中，它也能将研发的应用程序添加到运行时配置中，最终生成灵活的嵌入式应用系统映像。在包括了应用程序的 XPE 运行映像生成完毕后，就可将映像部署到目标设备上，来运行需要的应用程序功能。

但由于嵌入式目标设备往往没有开发主机上那么多的资源，一旦程序出现问题，调试起来非常麻烦，而且缺陷往往和具体的嵌入式设备硬件相关联，不能脱离实际的嵌入式设备运行，诸如此类的原因导致嵌入式目标设备的应用程序调试起来非常棘手。业内数据可看出调试的低效率，八成的嵌入式程序开发人员把近 50%的时间花在了程序调试上，项目延期多是严重 BUG 的调试查错效率低下所导致。故利用开发主机的资源，来实现 XPE 嵌入式设备的远程在线调试，对快速提升研发效率、快速定位解决问题都是很有意义的。

2 XPE 嵌入式开发中远程在线调试机理

所谓调试断点，是利用 X86 架构的 CPU 指令 INT3，结合相应的中断处理机制，使正在执行的程序暂停在 INT3 指令处，再通过访问 pdb 符号文件，以获得当前所在的源代码行，以及各个变量的值等信息。

在线调试分为本地在线调试和远程在线调试。可以使用 DEBUG 运行的方式，也可以使用附加 (Attach) 运行进程的方式。对于嵌入式开发的远程在线调试多使用 Attach 的方式，从而做到从研发人员的计算机上通过网络对嵌入式 XPE 上的程序进行调试。下文为陈述方便，做一个约定：

(1) 本地研发主机：研发人员用来开发的计算机，有强大处理器、大内存、大硬盘、网络等丰富资源，运行有 VS2010 集成开发环境，可以编程调试远程 XPE 嵌入式设备的应用程序。

(2) 远程 XPE 嵌入式设备：用来运行应用程序的嵌入式设备，各种资源都受限，要调试的有问题的应用程序运行在这上面。

3 本地研发主机上远程调试组件的安装与部署

第一步、在 32 位研发主机上安装 Visual Studio 时，默认情况下只安装 32 位的远程调试组件（除非安装过程中选择了 64 位远程调试组件，在 64 位 X86 研发主机上安装 Visual Studio 时，64 位和 32 位的远程调试组件都将安装）。远程组件通常情况下安装在【%ProgramFiles%\Microsoft Visual Studio 10.0\Common7\IDE\Remote Debugger\】目录下，32 位的在 X86 目录下，64 位的在 X86 目录下。目前 XPE 一般都是 32 位 X86 架构，所以只要安装的 X86 目录下有文件即可。

第二步、配置 Windows 防火墙（以 WindowsXP 系统为例，其他系统请参照）。在 Windows 防火墙对话框的【例外】选项卡中，用【添加端口】按钮来开放 135 端口（远程调试 DCOM 通信使用），用【添加程序】按钮来添加 Visual Studio 开发环境的主程序devenv.exe（位于【%ProgramFiles%\Microsoft Visual Studio 10.0\Common7\IDE\】目录下），以确保开放环境的主程序运行时能够打开 TCP 端口。

第三步、安全策略的配置。打开控制面板\管理工具\本地安全策略\本地策略\安全选项，将【网络访问】项设置为

来卡网出品 LAICAR.COM shop35833438.taobao.com

经典模式：本地用户以自己的身份验证。

第四步、用户和密码。远程调试涉及到权限问题，为简便配置，直接使用操作系统中系统管理员组里的用户，这里用administrator，而且密码要和远程 XPE 嵌入式设备上的administrator 一样。

4 远程 XPE 嵌入式设备上远程调试组件的部署

首先要将 32 位的远程调试组件，即【Microsoft Visual Studio 10.0\Common7\IDE\Remote Debugger\X86】目录下的所有文件，拷贝到远程 XPE 嵌入式设备上（不建议使用共享文件模式来运行远程调试组件，将会导致某些调试功能异常）。在调试远程 XPE 嵌入式设备上的应用程序之前，要启动远程调试监视器程序 msvsmon.exe。

其次若远程 XPE 嵌入式设备上启用了防火墙，要正确地配置。一是添加 135 远程调试 DCOM 使用的端口；二是在 Windows 防火墙对话框的“例外”选项卡中的“程序和服务”列表框中，选择“文件和打印共享”项目，然后用“编辑”按钮，在编辑服务的对话框中，选择“TCP 139”、“TCP 445”、“UDP 137”和“UDP 138”这 4 个端口；最后用“添加程序”按钮添加 msvsmon.exe 远程调试组件主程序。

另外注意 XPE 嵌入式设备上 administrator 的用户密码要和运行 Visual Studio 的本地主机上一样。

5 远程 XPE 嵌入式设备上的程序配置

运行远程调试组件的主程序 msvsmon.exe，选择远程调试监视器的“tools”菜单的“options”功能，并进行如图 1 所示设置：记录下“Server name”项中显示的远程嵌入式设备的主机名，后面设置会用到；身份验证模式选择 windows authentication；最长空闲时间，选择一个适合你实际情况的值。



图 1 远程调试组件配置

6 本地研发主机上的程序配置

工程属性选择 debug 标签页，如图 2 所示。

其中工作目录选择的是本地研发主机上该工程 debug 目录所在的路径；“Use remote machine”选项打勾，并输入之前记录的远程嵌入式设备的主机名；将“Enable unmanaged code debugging”和“Enable the Visual Studio hosting process”选项打勾。



图 2 本地主机的配置

7 远程在线调试的具体步骤

第一步，将本地研发主机上，该工程 debug 目录下的所有文件拷贝到远程 XPE 嵌入式设备上的目录下。例如本工程中，将 F:\SGS\MP\Console.Application\trunk\Projects\Console\bin\Debug\ 目录下所有文件拷贝到 XPE 嵌入式设备上 c:\control\ 目录下。

第二步，在远程 XPE 嵌入式设备上运行要调试的目标程序 c:\control\control.exe。

第三步，在本地研发主机上使用“debug”菜单中的“Attach to Process”功能连接远程 XPE 嵌入式设备。在弹出的对话框中进行如下选择：

“Transport”选项选择“Default”；

“Qualifier”选项选择“Adminisrator@OEM-FA796NIPVIZ”，注意 OEM-FA796NIPVIZ 是前面记录的远程 XPE 嵌入式设备的主机名。

单击“Attach to”按钮，对于.Net 程序，在弹出对话框的“Debug these code type”列表框中，选择正确的代码类型，然后根据.Net 版本的不同选择“Managed (v2.0,v1.1,v1.0) ”或者“Managed (v4.0) ”，非.Net 选择“Native”，如图 3 所示。

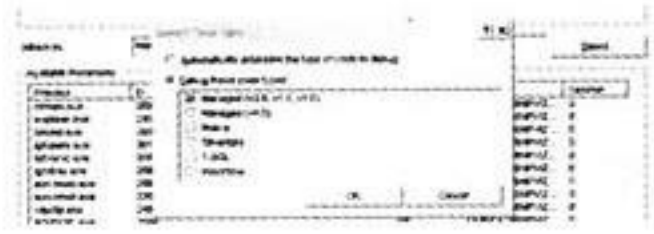


图 3 “Attach to”设置

最后在进程列表框中，选择要远程在线调试的程序，单击“Attach”按钮，就可以像在本地调试程序一样进行远程在线调试了，可以设置断点到源代码行上，查看各种变量的实时运行值，查看调用堆栈等。

参考文献

[1] Mark E. Russinovich and David A. Solomon with Alex Ionescu. 《Windows Internals Fifth Edition》, 2009.

[2] Microsoft. Windows Development Reference. http://msdn.microsoft.com/en-us/library/default.aspx.

（收稿日期：2013-03-26）

LAICAR.COM
shop35833438.taobao.com

Q 如何在 Windows 中提升应用程序权限与 UAC 交互

A UAC 是 User Account Control 的简称，这项新技术是 Microsoft 从 Windows Vista 操作系统开始引入的全新功能，目的是使用户账户运行在非超级管理员权限下，只有当确实需要修改系统信息时，才要求用户提供权限或管理员密码，以达到提升应用程序权限的目的。显然，这样做大大提高了系统的安全性，因为大多数用户习惯了使用类似 Administrator 这样的超级管理员用户来登录系统，而大多数情况下，用户实际上并不需要那么高的权限，UAC 的出现很好地解决了这一问题。

然而，UAC 也给用户编程带来了一些麻烦，尤其是习惯了 XP 及之前操作系统编程的开发人员，往往会忽视 UAC，而使开发出的程序在 Win7 系统下出现莫名其妙的错误，这是因为程序员往往习惯了向系统注册表或系统目录写入配置信息等类似需要管理员权限的动作，这在 Windows XP 下一般不是问题，但正如上文所述，在 Win7 下，这些动作默认会被系统拒绝，于是程序就无法正常运行了。下面分两种情况，介绍如何在 Win7 中提升应用程序权限来与 UAC 交互。

(1) 部分功能需要管理员权限

假设程序大部分只需要一般权限，只有某个操作时需要提升到管理员权限，这是很普遍的，比如程序只是在初始运行时需要写如配置信息到系统目录，或者程序在第一次安装时要写入注册表完成 Shell 安装等。下面笔者设计了一个 C# Project，在一个窗体上放置一个按钮，程序启动时不需要管理员权限，当用户点击按钮时，程序将告知用户点击此按钮需要提升权限，具体代码如下：

```
        private void button_elevate_Click (object sender, EventArgs e)
        {
            if (IsElevatedNeeded())
            {
                MessageBox.Show ("In order to use this function,the program will restart to gain elevated power", " info", MessageBoxButtons.OK, MessageBoxIcon.Information);
                ProcessStartInfo psi = new ProcessStartInfo ();

                psi.UseShellExecute = true;
                psi.FileName = Application.ExecutablePath;
                psi.Verb = "runas";

                try
                {
                    Process.Start(psi);
                }
                catch
                {
                    // 如果用户在 UAC 提示时拒绝,那么直接返回,程序也不必退出。
                    return;
                }
                Application.Exit();
            }
            else
            {
                MessageBox.Show ("power is already elevated", "info", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
        }
        bool IsElevatedNeeded()
        {
            WindowsIdentity id = WindowsIdentity.GetCurrent();
            WindowsPrincipal principal = new WindowsPrincipal(id);
            return ! principal.IsInRole (WindowsBuiltInRole.Administrator);
        }
```

这里做一些解释，当用户点击按钮时，代码通过 IsElevatedNeeded 方法判定当前进程是否有管理员权限，如果已经具备权限则可进行相关操作，否则，程序通过启动当前进程的一个新实例的方式来达到以管理员权限启动程序的目的，具体来说就是利用 ProcessStartInfo 类来构造一个新实例对象，当新实例启动后，老的实例自动关闭。这里还要注意 try...catch 的使用，这是因为当用户点击按钮后，Win7 系统的 UAC 模块会提示用户是否允许权限提升，用户可以选择拒绝提升，如果这样的话程序会进入 catch 模块，显然此时什么事情也不必做，而且老的实例也不需要退出。

(2) 所有功能需要管理员权限

还有一种情况就是程序的所有功能都需要管理员权限，比如安装程序等。希望是程序一启动时就自动提示用户权限提升，此时需要在 C# Project 中添加一种叫做“应用程序配置清单”的文件，可以在 VS2008 中添加一个 app.manifest 文件，该文件添加后其中有大量预置内容，要做的是添加以下行 <requestedExecutionLevel level=" requireAdministrator" uiAccess=" false" />于<requestedPrivileges>结点中，这样编译出的应用程序就能达到启动时自动提示提升权限的目的。

Win7 下的应用程序中某个按钮的点击操作需要权限提升，那么按钮上会有一个 Shield 图标，这是通过向按钮发送 BCM_SETSHIELD 消息做到的，代码如下：

```
        pr ivate void UACDemo_Load (object sender, EventArgs e)
        {
            bool blfNeedElevated = IsElevatedNeeded();
```


来卡网出品 LAICAR.COM shop35833438.taobao.com

TROUBLESHOOTING OF PROGRAM

```
            if (blfNeedElevated)
            {
                    this.button_elevate.FlatStyle = FlatStyle.
System;
                uint BCM_SETSHIELD = 0x0000160C;
                    SendMessage (new HandleRef (this.
button_elevate, this.button_elevate.Handle), BCM_SETSHIE
LD, new IntPtr(0), new IntPtr(1));
            }
```

沿用上面的示例代码，当应用程序启动时，如果判定需要权限提升，则向按钮发送消息来画出一个 Shield 图标以提示用户点击该按钮需要提升权限。

最后说明一下，要实现上述功能的应用程序，显然操作系统至少应该是 Windows Vista，而且要完成上述程序的编译，至少应该使用 VS2008 编译器。

（作者：李斌）

Q 如何快速清除 Word 文档中的重复文章

A 随着网络技术的高速发展，很多部门和单位开始有了自己的网站，内容服务自然就成了网站编辑的日常工作。然而，在众多的投稿和报送的信息中，有不少重复内容的稿件。单凭记忆的方式来筛选重复费时又费力，出错是在所难免的。不但影响工作效率，还会给单位造成一定的负面影响。下面就以 Word 2003 为应用平台，来给“网事”缠身的网站编辑排忧解难，快速批量地清除那些重复的稿件。

（1）稿件合并前的清除

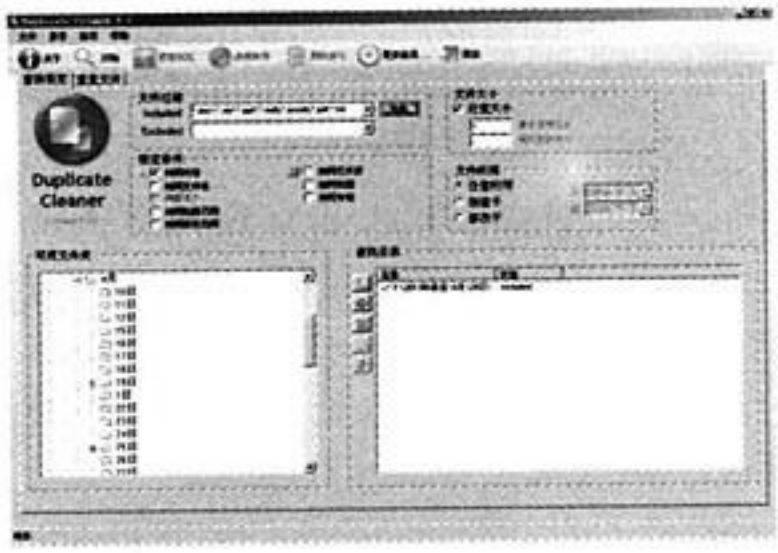


图 1

收完单位报送的稿件后，在整理之前进行甄别操作，把重复报送的稿件剔除出去，用一个小小工具即可搞定。网上关于这方面的工具有不少，这里推荐“重复文件查找工具 DuplicateCleaner 2.1 中文绿色版”这一款软件（下载地址 http://c1.cr173.com//soft2/DuplicateCleaner.zip）。首先，下载 DuplicateCleaner 2.1 程序，解压后安装到本地计算机中。运行该程序后，勾选程序窗口下的“相同内容”选项。单击程序窗口下的“选择”按钮，在“选择文件类型”对话框中选择“文档”选项后，单击“选择”按钮返回到上一窗口。在“可用文件夹”下选择要查找的文件来，单击“查找目录”下的“→”按钮，把文件夹导入到右侧的框架中（如图 1 所示）。然后，单击程序窗口的“开始”按钮进行扫描。等到扫描完成后，就会把所有重复的文件罗列在当前窗口的列表框中（如图 2 所示），随后把窗口的所有重复文件进行勾选，单击程序窗口的“删除选定”按钮，即可进行删除，这样就把重复的文件给删除了。

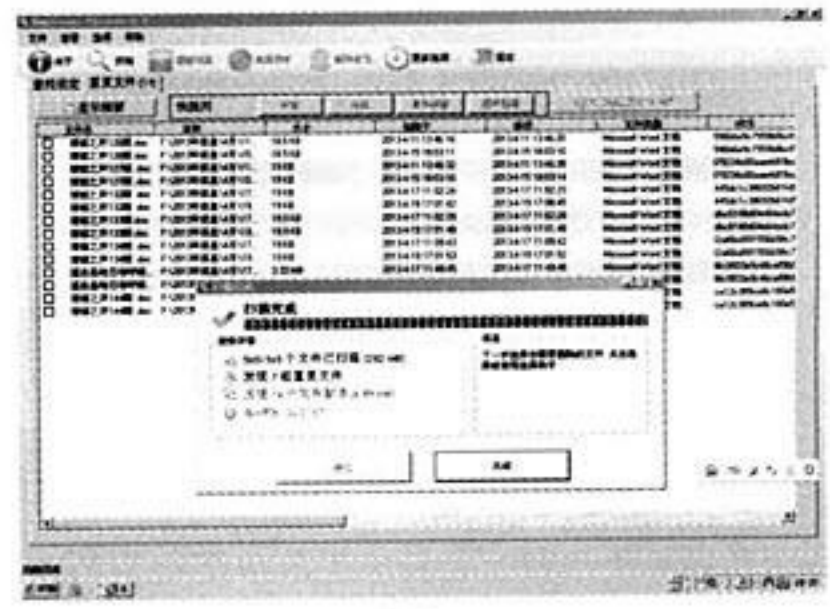


图 2

（2）稿件合并后的清除

1）使用通配符清除重复文章

下属部门其中的一项任务就是把收到文章（稿件）合并一个 Word 文档，报送给上级部门。当然了，重复信息是不允许出现的。下面就用 Word 通配符的方法来清除重复的文章。

① 文档段间空行。首先，打开已经整理好的 Word 文档（很可能还有重复文章在里面），依次单击 Word 窗口菜单栏的“编辑→查找和替换→”菜单，打开“查找和替换”对话框。接着，在对话框的“查找内容”后输入“^p”字符，“替换为”后输入“^p^p”字符后，单击“全部替换”按钮（如图 3 所示），就把整个文档中的所有段落自动空一行。然后，按下键盘的“Ctrl+Home”组合键，把当前光标移至文章开始处（这是由于 Word 默认是从光标开始处查找）。最后，再按一下回车键，使文章的开头也空一行（这一步狠关键，否则下面的操作将达不到预期效果的）。



图 3

② 清除重复的内容。按下键盘的“Ctrl+H”组合键，再次打开“查找和替换”对话框。接着，在对话框的“查找内容”后输入“(^13 [! ^13] @^13) (*) \1”字符（一定是半角字符，下面也是相同的），“替换为”后输入“^13\1\2”字符。单击

LAICAR.COM
shop35833438.taobao.com

对话框中的“高级”按钮后，勾选“使用通配符”。然后，单击“全部替换”按钮（如图4所示）即可进行清除操作。连续重复几次“全部替换”操作，只到替换数为0时。该操作对页数很多的文档需要的时间比较长，因为要执行循环的比较，所以大家要有一定的耐心。



图4

为了使大家知其所以然，对上述代码进行简单的说明。“^13”表示段落标记，“[!]”表示“非”，“[! ^13]”表示段落标记之外的任意单个字符，“@”表示一个以上的前一字符或表达式，它跟“[! ^13]”一起表示一个以上的段落标记之外的字符，“^13 [! ^13] @^13”也就是包含前后段落标记的、有任意字符的、非空的段落。而用小括号将其括起来“(^13 [! ^13] @^13)”，表示是将这一部分加以标识。替换框中的“\1”表示“(^13 [! ^13] @^13)”，{2,}表示2个以上完全相同的段落（括号括起来的部分）。

2）使用“宏”命令清除重复文章

① 设置宏的安全级（如果宏的安全级设置太高，宏功能是无法使用的）。用Word 2003应用程序打开整理好的文档后，依次单击Word 2003窗口的“工具→选项”菜单，打开“选项”对话框。单击“安全性”标签后，再单击其下的“宏安全性”按钮，打开“安全级”对话框。单击“中”单选框后，连续单击两次“确定”按钮完成设置。随后，重新启动Word 2003即可。

② 开启“宏”编辑器。首先，依次单击Word 2003窗口的“工具→宏→Visual Basic编辑器”菜单（或者使用快捷键“Alt+F11”直接打开），打开“Visual Basic编辑器”窗口。接着，在“工程”窗口下，右击“Normal”，在弹出的菜单中依次选择“插入→模块”菜单，打开“模块”代码窗口。接着，将“清除重复文章”的代码直接复制到右边的代码区（如图5所示）。然后，单击窗口的“保存”按钮进行保存。退出Visual Basic编辑器后，回到原来的文档窗口状态下。“清除重复文章”代码如下：

```
Option Explicit
  Sub 清除重复文章()
   Dim myStr As String, myArr () As String, aArr As
Variant
   Dim n As Long, book As Bookmark
    On Error Resume Next
    Application.ScreenUpdating = False
    With ActiveDocument
    myStr = .Content.Text
    myArr = VBA.Split(myStr, Chr(13))
    myStr = "|"
    For n = 1 To UBound(myArr)
      myStr = myStr & myArr(n – 1) & "|"
If VBA.InStr(myStr, "|" & myArr(n) & "|") <> 0 Then
      .Bookmarks.Add Name:="MyBM_" & n, Range:=.
Paragraphs(n + 1).Range
        .UndoClear
      End If
      Next
      For Each book In .Bookmarks
      book.Range.Delete
      .UndoClear
    Next
   End With
Application.ScreenUpdating = True
End Sub
```



图5

③ 运行“宏”命令。依次单击Word 2003窗口的“工具→宏→宏”菜单（或者使用快捷键“Alt+F8”直接打开），打“宏”对话框。选择“宏名”下的“清除重复文章”的宏，单击“运行”按钮即可对当前文档的重复内容进行清理处理（如图6所示）。



图6

来卡网出品 LAICAR.COM shop35833438.taobao.com



图 7

④ 设置快捷菜单，想要经常使用本功能，这样的操作就有点麻烦了，可以使用快捷菜单的方式。首先，依次单击 Word 2003 窗口的“工具→自定义”菜单，弹出“自定义”对话框。接着，单击“命令”标签，选择“宏”类别后，出现“Normal.模块 1.清除重复文章”的命令。把该命令值拖到工具栏中即可（如图 7 所示）。不过，这时在工具栏中显示的是模块的名称。随后，在“自定义”对话框的“命令”标签下，单击“重排命令”按钮后，打开“重排命令”对话框。单击“工具栏”后，再单击其后的下拉按钮，选择“格式”选项。然后，在“控件”列表中找到“Normal.模块 1.清除重复文章”并选中，再单击右侧的“更改所选内容”按钮，选择“命名”后，把文字改成“清除重复”即可（如图 8 所示）。这样，就长期安家 Word 窗口中，随点随用。



图 8

（作者：王昌红）

（上接第 87 页）

图 2 是国外专业软件测试 Word2003 文档正常时能够提取 Word 文本。



图 2　国外专业软件测试 Word2003 文档正常时能够提取 Word文本

(2) Word2003 文档损坏情况测试（Bad_ 信息化战争.doc）



图 3　该工具软件测试 Word2003 文档损坏时仍然能够提取 Word 文本

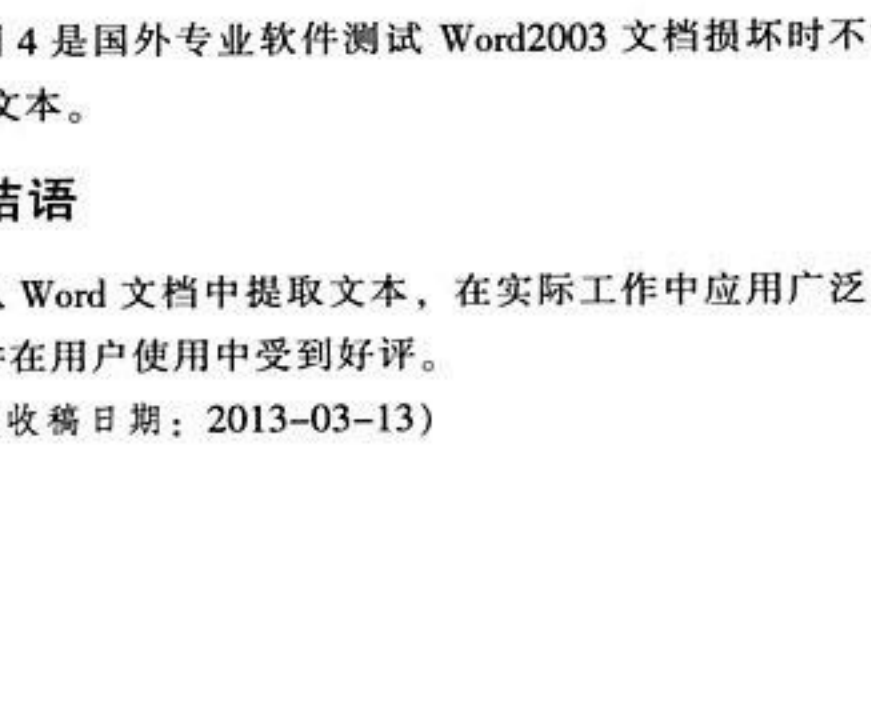


图 4　国外专业软件测试 Word2003 文档损坏时不能提取 Word 文本

图 3 是该工具软件测试 Word2003 文档损坏时仍然能够提取 Word 文本。

图 4 是国外专业软件测试 Word2003 文档损坏时不能提取 Word 文本。

5　结语

从 Word 文档中提取文本，在实际工作中应用广泛。该工具软件在用户使用中受到好评。

（收稿日期：2013-03-13）

LAICAR.COM
shop35833438.taobao.com

电脑系统维护经验与技巧

使用移动硬盘主板跳线如何优化

很多主板上都有选择 USB 供电的跳线，可供用户选择 5Vsb 供电或电源的+5V 供电。5Vsb 是提供给电脑启动时使用的，电脑在关机状态中仍有电力供应，可以让用户实现 USB 键鼠标开机等功能，不过这也会让接入电脑的移动硬盘在电脑关机后仍处于带电状态。而电脑中的+5V 供电是提供给主板、硬盘、光驱等设备使用的，供电能力非常强，一般能提供 20A 以上的电流，而 5Vsb 一般只能提供 2A 左右的电流。

因此，用户在选用了+5V 为 USB 供电后，可以增强每个 USB 接口的供电能力，可供用户接入更多耗电设备。同时，采用+5V 供电也能避免电脑关机后 USB 设备“不熄火”的情况出现。对移动硬盘起到保护作用。但是，每一块主板的布线并不相同，所以这一技巧对用户有较高的技术要求，用户在进行这方面的设置前，得事先了解主板说明书上关于“USB 供电跳线”的介绍，然后进行跳线的设置。

USB 3.0 移动硬盘为何会降速

要想真正体验 USB 3.0 所带来的高速度，一定要让主板端口、连接线缆、移动设备全部符合 USB 3.0 标准，而且驱动正常才行。如果使用了机箱的前置 USB 接口，而目前只有少数高端机箱的前置 USB 接口为 USB 3.0 标准，标志为蓝色 USB 接口，如果用户的前置 USB 接口并不支持 USB 3.0 标准，那么就相当于使用了一个 USB 2.0 的连接线缆进行数据传输，所以才会遭遇降速问题。建议在使用移动硬盘时，直接插在主板的接 USB 3.0 口上，并保证驱动正常就可以享受 USB 3.0 的高速了。

SATA 数据线如何区分好坏

若有两根 SATA 数据线，第一根上面写着“30AW G E173433 - P AWM STYEE 2725 80℃ 30V VW -1 COMAX”，第二根上面写着“AWM E174881 STYLE 21149 30V 80℃ VW -1 26AWG TECHPOINT”，AWG（American Wire Gauge）为美国线规，是一种区分导线直径的标准，AWG 前面的数值（如 26AWG、28AWG）表示导线形成最后直径前所要经过的拉伸孔的数量，数值越大，导线经过的拉伸孔就越多，导线的直径也就越小，反之，AWG 值越小就表示导线直径越粗。如果单从这两根 SATA 线标注的 AWG 值来判断第二根数据线相对更好些。

如何屏蔽硬盘坏道

通过隐藏分区的方法来屏蔽硬盘坏道，其实操作起来并不复杂，直接使用系统内置的磁盘管理工具即可，不需要使用第三方软件。不过因为操作系统一般是安装在硬盘的第一个分区内的，所以用户得用 U 盘或光盘启动到 PE 系统（一个只拥有最少核心服务的 Mini 操作系统），或将硬盘接到他人的电脑上，在另外的操作系统上为该硬盘“动手术”。

具体方法是：右击“我的电脑”→“管理”→“磁盘管理”，在窗口右侧找到自己的硬盘（注意别选成其他的硬盘），将其中所有分区都删除掉，这时开始建立第一个主分区，容量为 35GB，紧接着再建立另一个主分区（记住是主分区），用作以后的系统分区，容量自己定。剩余的空间可以全部划分为逻辑分区，最后一步便是将第一个 35GB 的分区删除。注意，在重新分区前别忘了先备份好自己的数据。

怎样使用 IDE 转 SATA 转接卡

IDE 转 SATA 或 SATA 转 IDE 转接卡是独立于其他硬件的，所以正常情况下接好后是不需要作其他设置而直接使用的。不过使用此类转接卡需要注意以下几点：

首先，转接卡本身是需要供电的，其电源接口就是提到的那个 4PIN 接口，但这仅仅是为转接卡供电，硬盘同样是需要接上电源线的;第二，现在很多此类转接卡都是支持双向转接的，既可以将 SATA 硬盘接到 IDE 接口上（此时转接卡接在不板 IDE 接口上），也可以将 IDE 硬盘接到 SATA 接口上（此时转接卡接在 IDE 硬盘上），因此转接卡上会有两个不同的 SATA 接口，分别用于这两种情况，注意一下是不是接错了。如果用户用的光驱是 IDE 接口的，那么可以让 IDE 硬盘跟光驱共用一根 IDE 线，只要设置一下主从跳线就好了。

固态硬盘由哪两部分组成

随着固态硬盘技术的发展和闪存价格持续走低，SSD 固态硬盘的普及已是大势所趋。其“轻薄的特性，极小的延迟时间，强悍的 4KB 随机读写，快速的 IOPS（每次读写操作次数）”都是机械硬盘无法企及的。正因如此，吸引众多 PC 用户选择 SSD 给电脑提速。

SSD 固态硬盘最重要部件由两部分构成，一是主控芯片，它负责控制闪存的识别和读写；二是闪存芯片，在固态硬盘中充当存储空间的介质。不同品牌的 SSD 所用的主控芯片不同，

来卡网出品 LAICAR.COM shop35833438.taobao.com

MAIL TO THE DOCTOR

这就关系到SSD固态硬盘的使用速度和使用寿命。

目前主流的固态硬盘所采用的主控与闪存大部分来自SandForce、Marvell、Intel、Indilinx、Samsung，TOSHIBA等唯数不多的国外知名厂商。

使用SSD要注意哪些技巧

SSD相对传统机械硬盘，它还是比较娇嫩的。所以需要掌握一定的SSD使用技巧，才能更好地使用SSD。

（1）使用官方推出的软件定期优化有助于维持SSD的性能和健康，比如Intel用户可以使用Intel SSD Toolbox（其他品牌的SSD也可以使用这个工具）。

（2）少用测试类软件。一般来说经常进行写性能测试对SSD的健康不利，所以写测试不需过于频繁。HD Tune、ATTODiskBenchmark、CrystalDiskMark、AS SSD Benchmsrk、Crystal DiskInfo这类软件的写测试都少用为好。

（3）如无必要，不用组建RAID，组建RAID会丧失Trim功能，在一定程度上会影响磁盘性能，一般来说不需要为了让持续读写成绩好看一些而对SSD进行组建RAID操作。

（4）不要把SSD作为下载盘，当然把下载软件（迅雷、电驴等）安装在SSD中使用起来启动软件的速度会明显提高，但不要把文件也下载到SSD中。

（5）SSD虽然在使用寿命上与传统机械硬盘有一定差距，但用户也不用过度小心，部分用户奉行“只读不写”，将系统和软件的交换、缓存文件存放在机械硬盘中，虽然这样对SSD的健康有好处，但SSD本来就是为了解决读写大量零碎文件的延时而诞生的，移动缓存文件就体现不出SSD的性能了。而且，目前的SSD产品大多具有三年以上的质保，其耐用程度还是比较让人放心的。

（6）SSD毕竟容量相对较小，因此如何合理优化空间也是一门学问。定期使用软件将不需要的缓存文件、系统垃圾清理出去保持SSD拥有一部分预留空间，这样对其性能的保持也有很多好处。

（7）部分用户过于担心SSD在存储数据方面的安全性，其实大可不必。通常来说SSD发生故障的几率并不大，而且大多SSD发生故障以后磁盘中的文件仍然可以被读出。当然对于那些比较重要的文档、数据，用户也要做好备份工作，将其复制并放在可靠的机械硬盘中更加保险。

基于DirectUI架构的中海油软件平台的用户体验设计与实现

中海油是中国海洋石油总公司的简称，是1982年成立的国家石油公司。经过几十年的不断发展，中海油拥有上百个油田工作系统，而每个油田系统均配有功能强大的软件系统作为它的支撑。每套软件系统的不断升级和维护让它们的程序变得更加复杂。由于这些软件都是从各种作业实践中逐步提炼而成的，从一开始就缺少统一的规划和全局的用户体验考虑，所以导致了一套系统一个样。给油田作业的工程师们带来了极大的操作难度。目前，中海油的软件系统的用户体验方面的问题已经严重影响到了海上作业的工作效率。中海油集团的上上下下也对用户体验问题高度关注。

UIPower在充分了解中海油的后台系统的需求后，建议中海油采用UI设计+UI开发的整体UI解决方案。

UI设计阶段：进行了用户研究、交互设计、用户体验测试和视觉设计等方面的工作。首先解决了软件难用的问题，对软件中的各种窗体进行了重新规划，详细分析了各种控件在实际使用中的频率。接着提出中海油作业软件的整体交互设计规范。从规范上确保未来的各个软件系统保持一致的操作习惯。通过对新的交互设计严格的实验室测试来检验它的科学合理性，去除里面不合理的地方，不断迭代最终形成易上手、易操作的软件交互模型。UIPower的视觉设计师根据多年的经验对软件进行整体风格、色调、质感等的创意，最终形成在视觉上具有高度专业性的软件高保真设计稿。

UI开发阶段：对于UI改造项目来说，设计出漂亮的界面其实已经很困难了，而要将专业的UI设计真正实现出来则是难上加难。原因是客户端各个开发工具对UI的实现均支持不足，无法满足各种新特效新质感的窗口或控件的交互和视觉呈现。一般来说，要实现高保真的效果图程序需要花费半年甚至更多的人月数。然而，现实却无法允许这样的界面开发进度。UIPower一直采用自主研发的DirectUI产品作为他们的界面开发的基石。DirectUI可以将一个软件的所有界面在1周内全部实现出来，而且对界面的效果做到没有损失，UIPower对项目验收的标准是“像素级比对”，即在最后的成型程序与高保真的视觉设计稿通过屏幕的逐像数比对，只要有一对像素的值不一样即认为是不通过验收。正是基于这样的验收标准，UIPower在业内迅速成为国内大型企业争相合作的对象。

经过2个多月的整体改造，中海油集团内的几款重要的软件系统，均已成功上线，并正式投入了使用。从集团上上下下的相关人员对本次的UI改造成果表示了高度的认可和极大的满意。

UIPower于2004年成立至今，一直以“让天下没有难做的界面”为使命，先后服务于华为、中兴、盛大网络、中国移动、中国电信、铁道研究院等国内大型企业。UIPower的阙总在接受采访时说：“未来UIPower一直致力于界面产品的研发与升级，全面助力中国软件相关企事业单位，提高中国软件产品在国际市场上的竞争力，实现软件强国的梦想！”。

LAICAR.COM shop35833438.taobao.com



书名：SAP 内存计算——HANA
ISBN：978-7-302-32357-0
定价：68.00 元
作者：胡健 和铁东 等著

本书是 SAP 官方权威指南，深度剖析了 SAP HANA，本书由农夫山泉股份有限公司前首席信息官胡建及首席方案架构师、SAP 大中国区方案与架构设计部总监和铁东著作。

全书以通俗易懂的语言、翔实生动的实例，通过结合包括农夫山泉 SAP HANA 实施全过程在内的相关 HANA 应用场景与案例，为读者详细解读 SAP HANA 这一业界领先的内存计算技术从原理到架构、从功能到实现的全方位信息。全书共分为 7 章，内容涵盖 SAP HANA 简介、安装 SAP HANA 数据库系统、SAP HANA 的数据复制、SAP HANA 的数据建模、SAP HANA 的数据表现、SAP HANA 系统管理以及 SAP HANA 的实战场景等内容。



书名：Visual C++ 2012 入门经典 (第 6 版)
ISBN：978-7-302-31900-9
定价：98.00 元
作者：(美) 霍尔顿 (Horton, I.) 著

本书是世界著名的计算机畅销书作者、IBM 多语言编程资深顾问、超级畅销书《C 语言入门经典（第 4 版）》作者、编程导师霍尔顿（Ivor Horton）最新作品，同时也是国内领先的 Visual C++ 2012 著作，本书引领无数程序员进入编程殿堂，是学习 Visual c++2012 经典指南

Ivor Horton 的书能够占据市场的一个原因是：他用其独特、易于理解的教程风格来介绍各个主题，无论是编程新手，还是经验丰富的编程人员，都很容易理解。现在在这本修订的指南中，他全面介绍了标准 C++语言和 Visual C++2012。循序渐进的编程练习、实用的示例和清晰的解决方案，带领读者学习 C++开发的来龙去脉。学习完本书，您就会牢固掌握使用 Visual C++ 2012 创建实际应用程序的技术。



书名：信息安全原理与实践（第 2 版）
ISBN：978-7-302-31785-2
定价：68.00 元
作者：(美) 斯坦普 (Stamp, M.) 著

本书作者在信息安全领域已有将近 20 年的经验，是美国国家安全局密码分析专家，本书既透彻讲解复杂的安全问题，又不陷于繁琐的细节，是面向 21 世纪的信息安全专业指南。

信息安全是一个快速发展的领域。着眼于极富时代感的安全议题，涵盖涉及宽泛的一系列新鲜信息，通过聚焦于现实世界中的生动实例，并采用一种面向实践的信息安全讲述方法，这本书围绕如下 4 个重要主题进行组织并展开：密码学技术、访问控制、协议、软件安全，在本书中，特别引入了一些比较新的内容，其中涉及的安全主题包括 SSH 协议和 WEP 协议、实际的 RSA 计时攻击技术、僵尸网络以及安全证书等。



书名：RHCSA/RHCE Red Hat Linux 认证学习指南 (第 6 版)：EX200 & EX300
ISBN：978-7-302-31712-8
定价：98 元
作者：(美) 江 (Jang, M.) 著

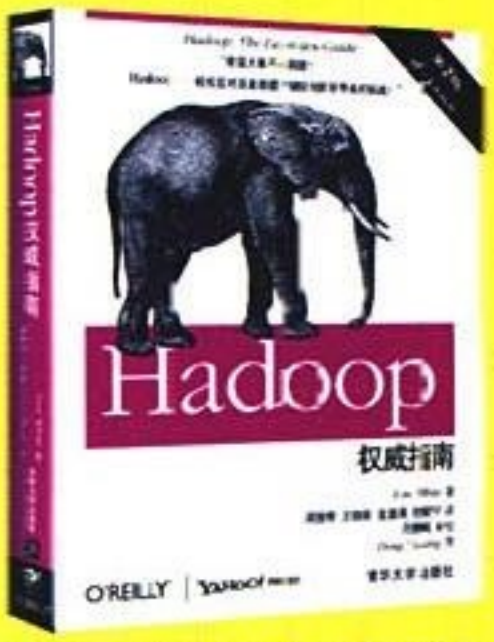
本书是亚马逊畅销榜图书，经过教学检验的 IT 培训和考试准备工具，100%覆盖——300 多道考试真题，覆盖所有 RHCSA 和 RHCE 考试。

本书不仅包含数百道复习题、完全覆盖基于性能的所有要求，而且包含你想知道的一切——为您详细介绍如何准备这两个极具挑战性的考试。100%完全覆盖 EX200 和 EX300 的所有正式目标考试准备检查清单——如何完成该清单上的所有目标，考试就没问题每章的考试部分突出显示主要考试主题用于快速复习的小练习 100 多个实验问题——两套完整的基于实验的 RHCSA 考试题和两套完整的基于实验的 RHCE 考试题——与真实考试的格式、风格、主题和难度相一致。

来卡网出品
LAICAR.COM
shop35833438.taobao.com

高端汇 技术前沿

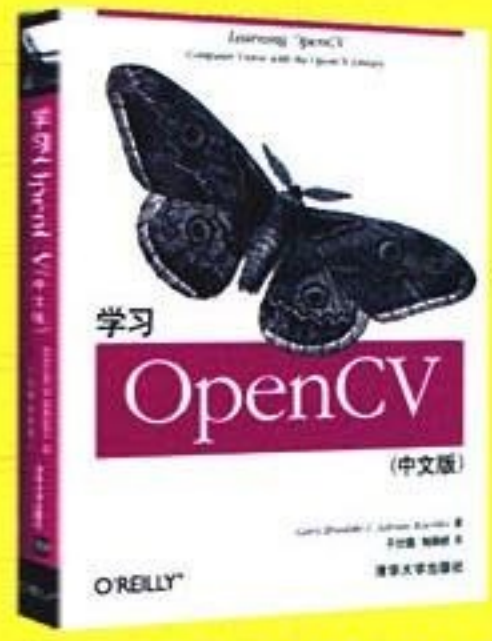
清华大学出版社
http://www.tup.com.cn



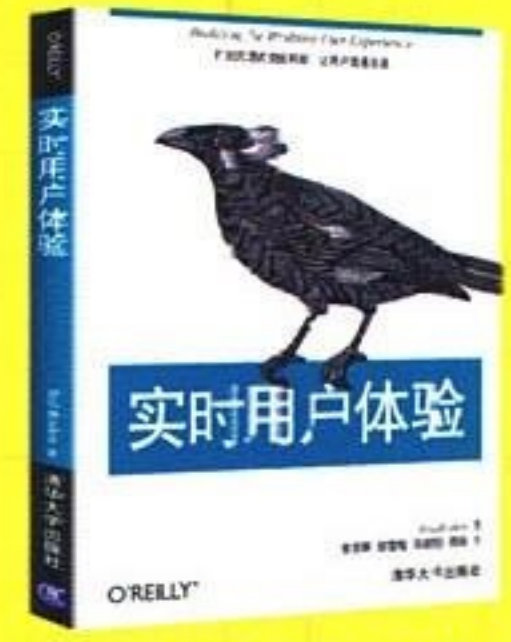
书号：978-7-302-25758-5
定价：89.00元



书号：978-7-302-29098-8
定价：89.00元



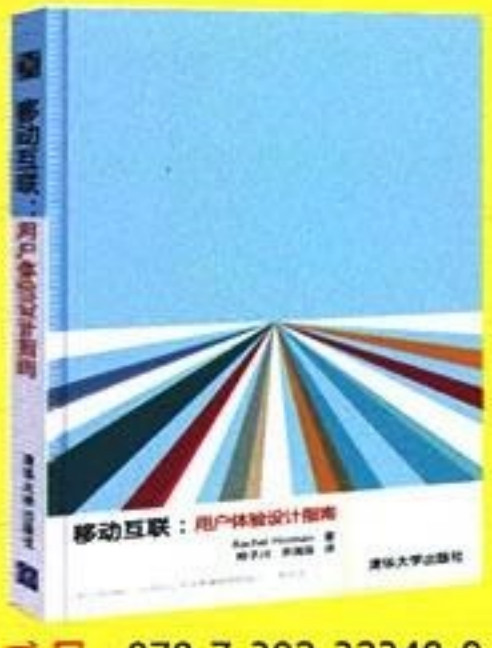
书号：978-7-302-20993-5
定价：75.00元



书号：978-7-302-29455-9
定价：59.00元



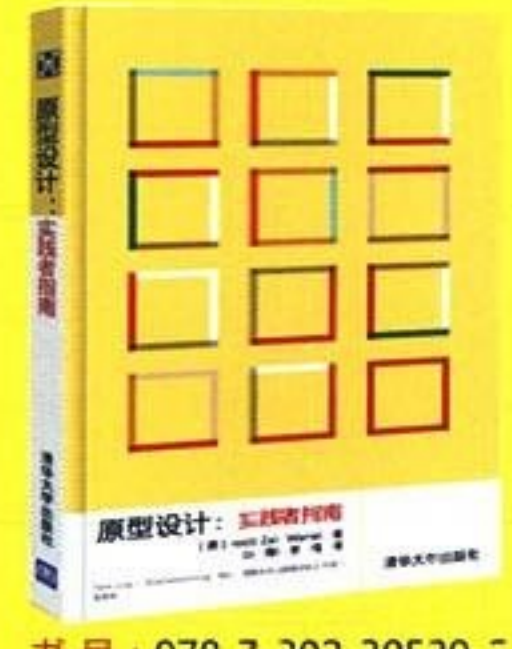
书号：978-7-302-31063-1
定价：69.00元



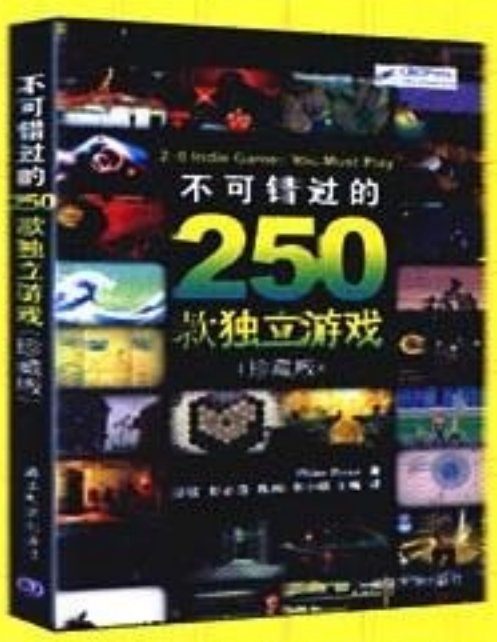
书号：978-7-302-32348-8
定价：79.00元



书号：978-7-302-31186-7
定价：59.00元



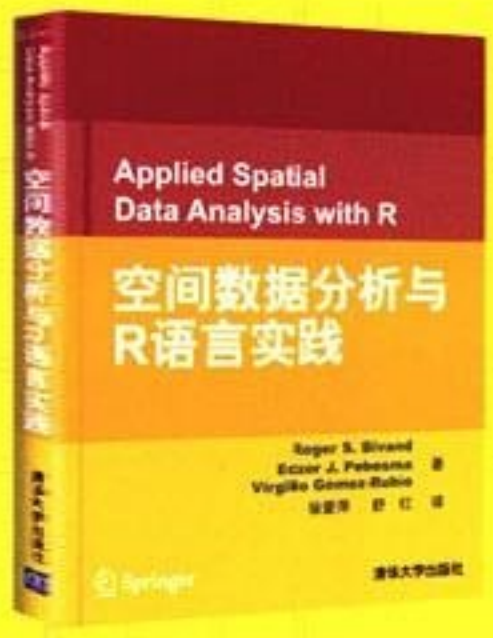
书号：978-7-302-30539-2
定价：59.00元



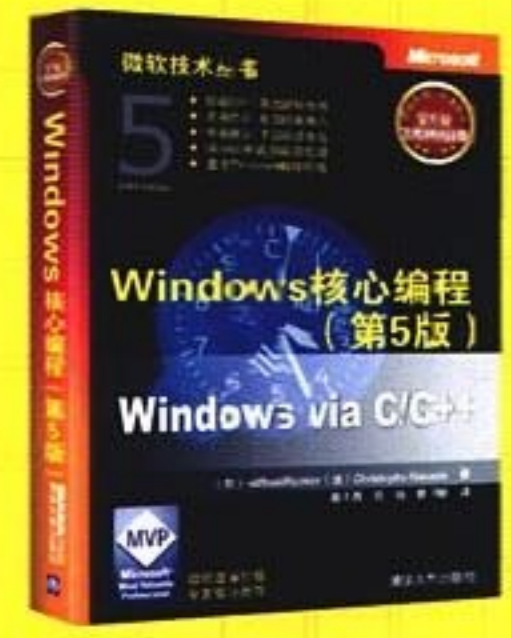
书号：978-7-302-31460-8
定价：49.00元



书号：978-7-302-30609-2
定价：130.00元



书号：978-7-302-30235-3
定价：79.00元



书号：978-7-302-18400-3
定价：99.00元



书号：973-7-302-30639-9
定价：69.00元



书号：978-7-302-31185-0
定价：79.00元



书号：978-7-302-31616-9
定价：79.00元



书号：978-7-302-31187-4
定价：75.00元

shop35833438.taobao.com

汇主流编程语言　精选典型编程案例

聚宝贵编程经验　揭示编程技术诀窍



《电脑编程技巧与维护》杂志社　编著

定价：79元

订阅方式：

汇款地址：北京市海淀区长春桥路5号6号楼1209室　　收款人：电脑编程技巧与维护杂志社　　邮编：100089

杂志社官方淘宝店：http://comprg.taobao.com　　电话/传真：82561314

E-mail:zzsfx@vip.sina.com　　QQ:565699495

汇款如未注明所购买数量和邮寄地址，请与杂志社联系。

来卡网出品
LAICAR.COM
shop35833438.taobao.com